Documentation for the swtor add-on for tails V 0.52

Authors : swtor00@protonmail.com

Date : 01.02.2021
License : GPL 2.0

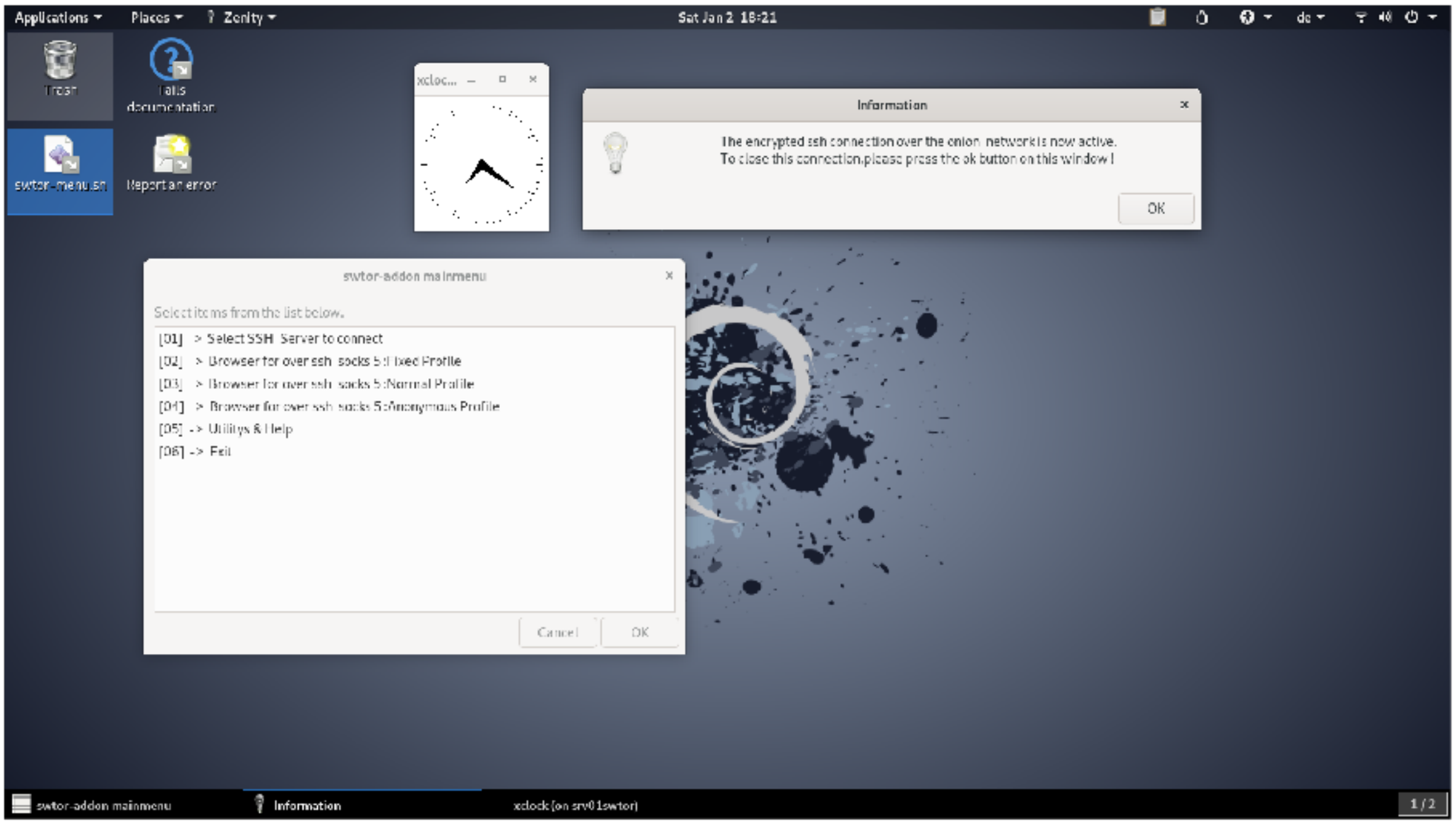


Table of contents

Introduction

This documentation describes how to install and use this add-on for Tails-Linux 4.14 or later. You may ask yourself, why do I need such a add-on for Tails ? The Tails Linux system already protects your privacy by using the Onion-Network on every startup. It is also true that a Tails system already makes a very good job to hide your true identity and real IP-Address to the websites you are visiting.

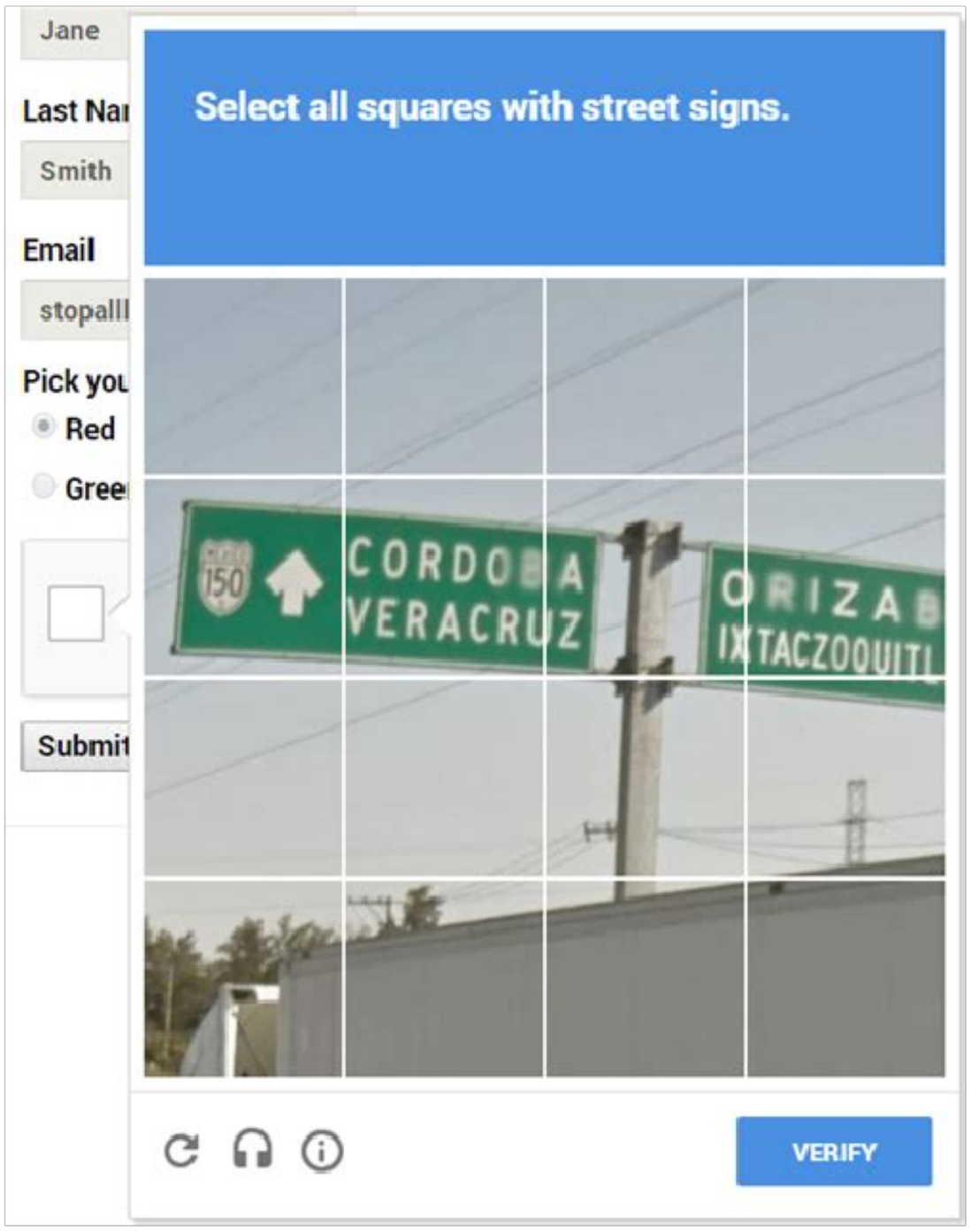
Although, it seems many Tor users are having difficulties surfing and navigating the regular World Wide Web (sometimes called Clearnet Internet) , as some websites have set up discriminating rules against people who are using the Tor Browser to browse the web. At any specific moment in time when we are using Tails or the TOR-Browser in general, our personal IP packets are sent through the Internet crossing 3 different Nodes to hide the origin where the packets came from.

The last of this 3 nodes so called “Exit-Nodes” of the Onion-Network can easily and instantly be detected as soon as someone queries with any remote IP Address from the list below.

https://check.torproject.org/cgi-bin/TorBulkExitList.py?ip=1.1.1.1

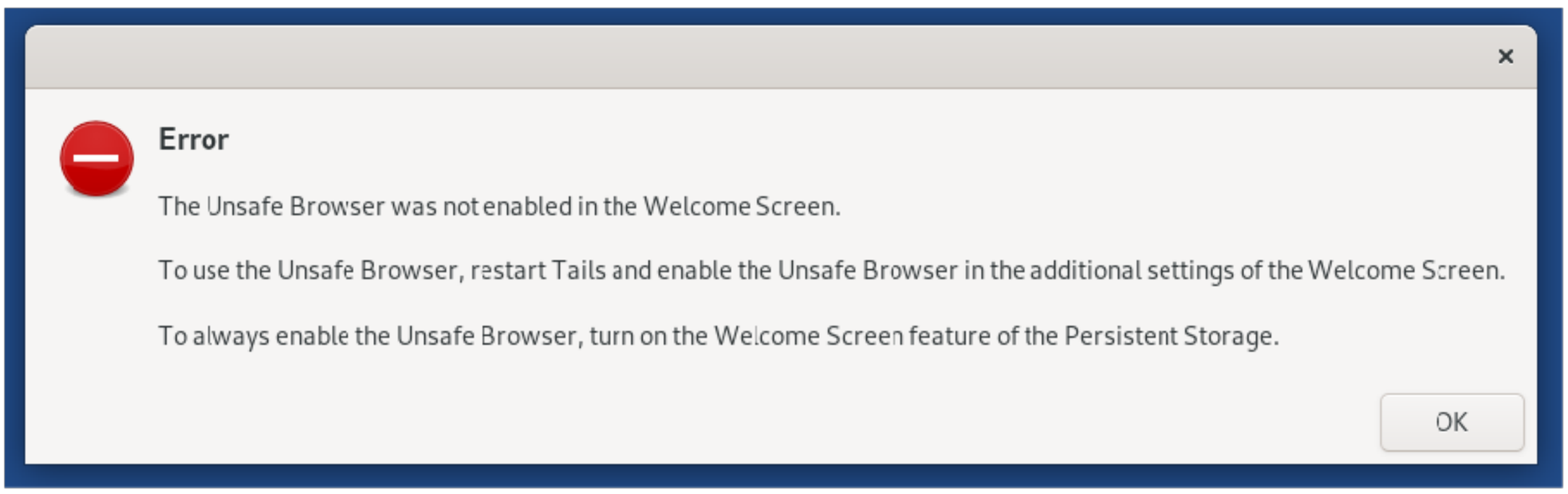
Therefore it is no longer a hidden secret to any website, that we are using the TOR-Browser or even Tails and they will often run this act of “captcha terror” against regular users of the TOR-Browser.

or something similar like this ...



Alarmingly, the number of websites that completely block or run the “captcha-terror” against regular TOR-Users is growing in numbers day by day, to the detriment of Tor’s usefulness.

Integrated inside every Tails installation on a CD or USB medium, there is still the so called “Unsafe Browser”. Because this “Unsafe Browser” doesn’t make any use of the Onion-Network at all, all data communications from this browser can easily be logged by your Internet Service Provider (ISP) or your government. After Tails 4.14 was released,it was possible to completely prevent the starting of the unsafe Browser inside of Tails.



By the default settings from greeting screen of tails 4.14 or higher, the start of the “unsafe Browser” is not longer possible until a user set the default value “not starting “ to “allow starting”.

It may make sense to use the “Unsafe Browser” inside Tails solely to browse a captive portal like it is used in many public WiFi networks, but not really for more than this simple task.

https://tails.boum.org/contribute/design/Unsafe_Browser/

Prior to activate the use of the unsafe Browser in the greeting-screen , please keep the following in mind.

IF YOU DON’T NEED THE UNSAFE BROWSER …
DON’T ACTIVATE IT AND USE IT !!!!

Some smart people even try to start a VPN connection with OpenVPN or something similar to hide the fact that they are using the Onion-Network, which can be blocked by any website or even an ISP.

Apart from the fact that you are only able to use a single TCP port for the very popular OpenVPN communication (all possible UDP ports inside of Tails are blocked by default), it produces many more problems than it would solve. In the endless debate about the use of a Virtual Private Network inside of Tails, I recommend to read the following URL.

https://tails.boum.org/blueprint/vpn_support/

“Similarly, we don't want to support VPN’s as a replacement for Tor since that provides terrible anonymity and hence isn't compatible with Tails' goal.”

What we want

Tails -> Tor -> VPN

USE CASES

1. Access services that block Tor.
2. Reach a local resource on a VPN that is not accessible in any other way.
3. Reach a VPN non-anonymously (e.g. your account is tied to you URL) while only hiding your geo-location, which may be the only thing you need in some situations. (Maybe invalid since this is not part of the PELD spec (yet?) AFAIK.)

SOLUTION

The easiest solution for this case 1 (which we feel is the most important one for this Tor/VPN setup) is to use an SSH connection with the Dynamic Forward option. The newly created SOCKS port can be used to have a fixed outgoing IP address. We may document how to use this in an "unsupported, advanced users only, “may kill kittens" part of the documentation.

Now, exactly right here, my special add-on for Tails enters the game called “Privacy,” and provides some very useful functions for the many Tails users out there :

- Use of an encrypted SSH-connection to a Remote Host and building of a local socks5 proxy. Even the traffic that is sent over the so called “Exit-node” of our communication is still encrypted until it reaches the destination SSH-Server. When the connection packets leave the SSH-Server to any external website as a example, the packets are not longer protected by SSH and would look like a ordinary traffic from a Desktop Computer running Linux.

- All SSH traffic is encrypted and routed over the Onion-Network, as long you are using an SSH server anywhere on the Internet. If you are using a SSH Server at your own network at home, only the internal connection from you tails system to the other internal ssh server is encrypted. And I guess only, this is not the way you would like to go to hide the fact that you are using tails ...

- A local Browser (Chromium) with three different profiles that can be used to visit TOR unfriendly websites like Google, and many other websites that would block regular TOR users like second class Internet users. All three predefined Chromium profiles are protected against multiple actions like WebRTC and trackers in general.

- All local DNS resolution traffic on UDP Port 53 is routed over the local socks5 proxy to the remote ssh-server . This means you never contact you local DNS server, like you would do with the unsafe browser.

- For any particular website, anywhere on the Internet, it is no longer possible to detect that we are using the so called “Onion Network” to hide our personal information or IP address at all. All the traffic that the owner of the website can analyze, is coming from the regular public IP address of the remote SSH server we are currently connected to. The fact that we are using the “Evil Onion-Network” to establish our SSH connection, is not visible to the websites we are visiting.

One huge problem still remains with using the Tails system to contact onion-unfriendly systems. It makes no difference what kind of bypass protocol is used to hide the fact that we are using Tails or the onion-network in the background. This so called bypass systems could be done by one of the three following techniques.

- A remote SSH-Server using a local socks5 connection port. This is the connection type we are using inside of the add-on and is the recommend way of the developers of tails.

- OpenVPN Server with a single TCP connection

- Proxychains

All other currently wildly used VPN solutions (LL2TP / IPSEC as a example) that would rely on a active UDP connection, would not work inside Tails because UDP protocol isn't supported at all.

By now follows a little Warning about using any of the bypass systems inside of tails including this add-on itself !

At this point you would like to raise awareness about the trust you are willing to give to a foreign host system and his administrators or users as well, who could easily read your complete communication that would be sent through to the foreign server you haven chosen to connect.
If you would like to use any of the 3 above bypass techniques, please don't underestimate the control they have over you in the exact moment in time that you are using their servers. Almost any VPN Service provider worldwide out there on the Internet make claims on their websites with cheap marketing statements like the following ones:

"We don't log anything !"
"We don't spy our users !
"We protect the privacy of our customers !"

Some very interesting articles about the “no-log policy” of some very popular VPN providers can be found here.

http://www.theregister.co.uk/2011/09/26/hidemyass_lulzsec_controversy/

https://torrentfreak.com/ipvanish-no-logging-vpn-led-homeland-security-to-comcast-user-180505/

https://www.techworm.net/2018/08/reasons-why-shouldnt-use-free-vpns.html

https://www.techrepublic.com/article/16-popular-vpns-leak-your-data-heres-the-full-list/

If you read the above articles carefully, you may come to the final conclusion that you shouldn’t give your personal trust out to the first person or company who offers you a VPN or SSH account for free. Even if you pay for a service with a monthly fee, there is no guarantee that you aren’t being “tracked” by this VPN server or any other active users of the remote system.

To start off with, there are three simple SSH scenarios possible, in which you can use my add-on to surf the Internet without being blocked from any website you would like to visit.

Prior to show 3 possible working scenarios with my add-on, we have to talk about the dangers using it at all. From all next described scenarios, you only should use the last presented scenario if possible.

- Never do a login with a username or email-address that you ever used on the normal Internet anytime. This kind of internet connections could be easily tracked back to you as a person or a company. This includes of course any kind of login for email, twitter or Facebook. If you need access to any website with your real email while you are using tails, use a other computer of your home network or do a reboot to using Windows or Linux.

- Only use my script to solve problems with tor-unfriendly sites at all and if possible every time use the more secure builtin tor browser of tails to connecting it.

- Never using any website simultaneously inside of the TOR-Browser a the Chromium Browser used by the script.

- Don’t use your mobile phone for a 2-Step verification while you are using Tails or TOR-Browser.

- Don’t operate on a user account created for Tails only, outside of Tails.

- Don’t post any personal information that can be tracked back to you.

  - Emails-Addresses that you ever used outside of tails
  - Credit card Information
  - Phone Numbers

- Don’t use Google for searching, even if you are connected to a remote server over SSH, that could use the Google Search Engine. Better use the following alternatives :

  www.startpage.com

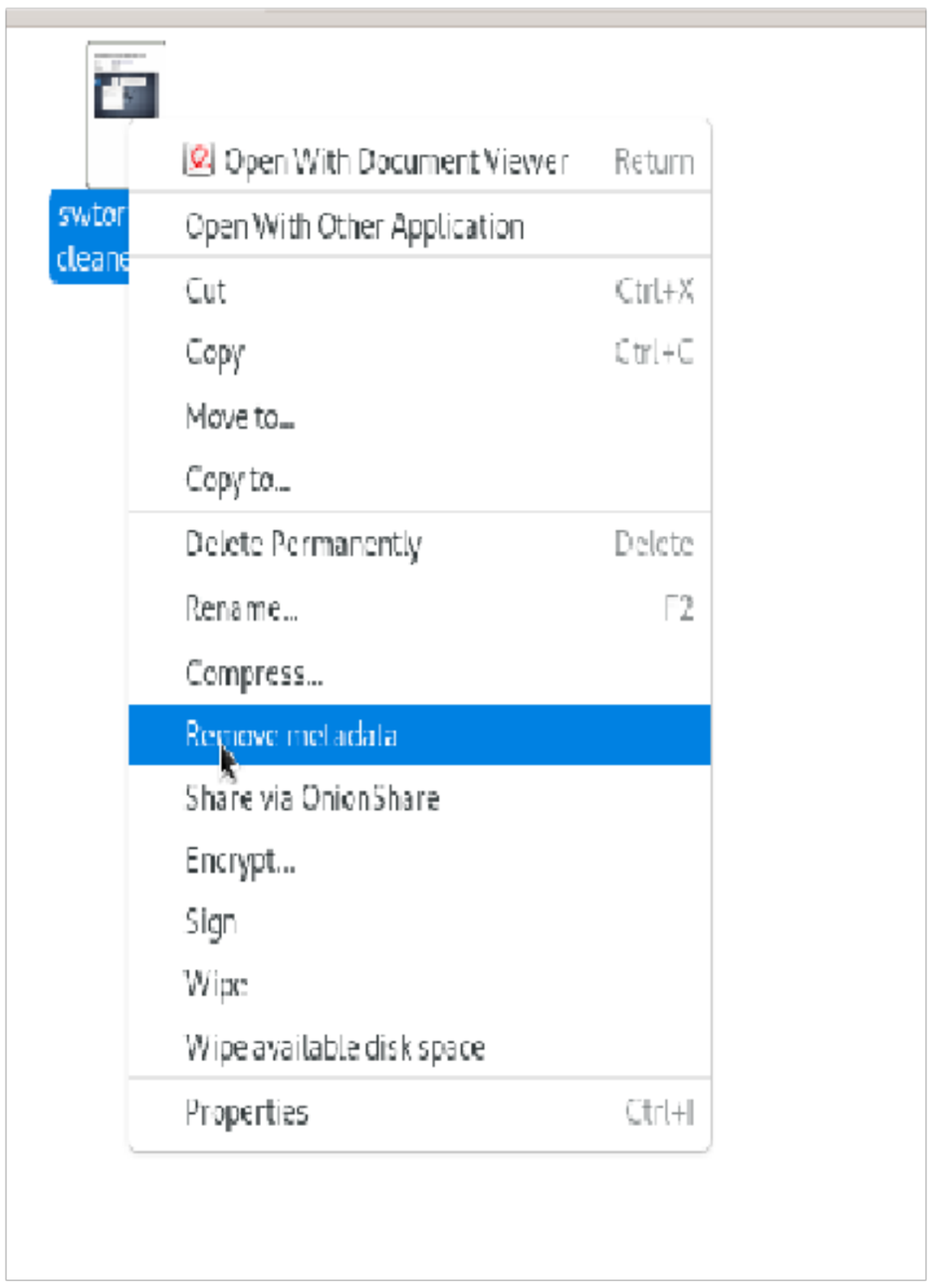
  or

  www.duckduckgo.com

- Never post any data like photos or documents without uncleaned meta data first. Inside of tails it is possible , to clean up all meta-data with a single click over the file-manager.

  https://tails.boum.org/doc/sensitive_documents/metadata/index.en.html

On the following screenshot, you see how easy it is to remove all meta-data from a existing file.

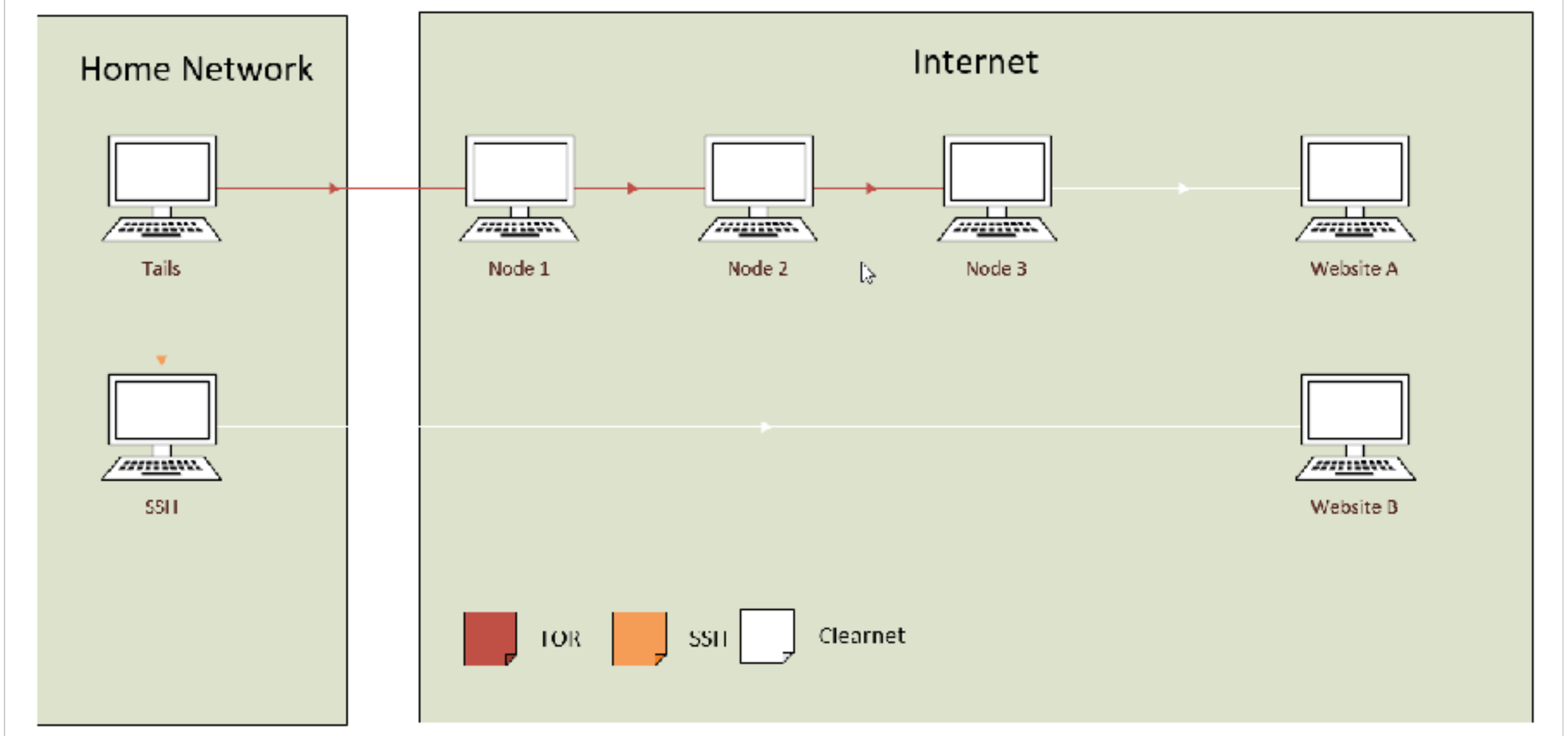


1.0 Using your own SSH server inside your own Network at home

For this simple and and not really recommend scenario, you need at minimum, a second computer with Linux or Unix running on your own network at home.
For this SSH-daemon you could use, for example, a simple Raspberry-Pi, or of course, any other computer with an SSH daemon would work as well.
This could be implemented with a Linux System like Debian or many others without any problem. For a simple example to build a SSH-server on a Raspberry-Pi I would recommend the following URL.

https://www.raspberrypi.org/documentation/remote-access/ssh/

Scenario 01:

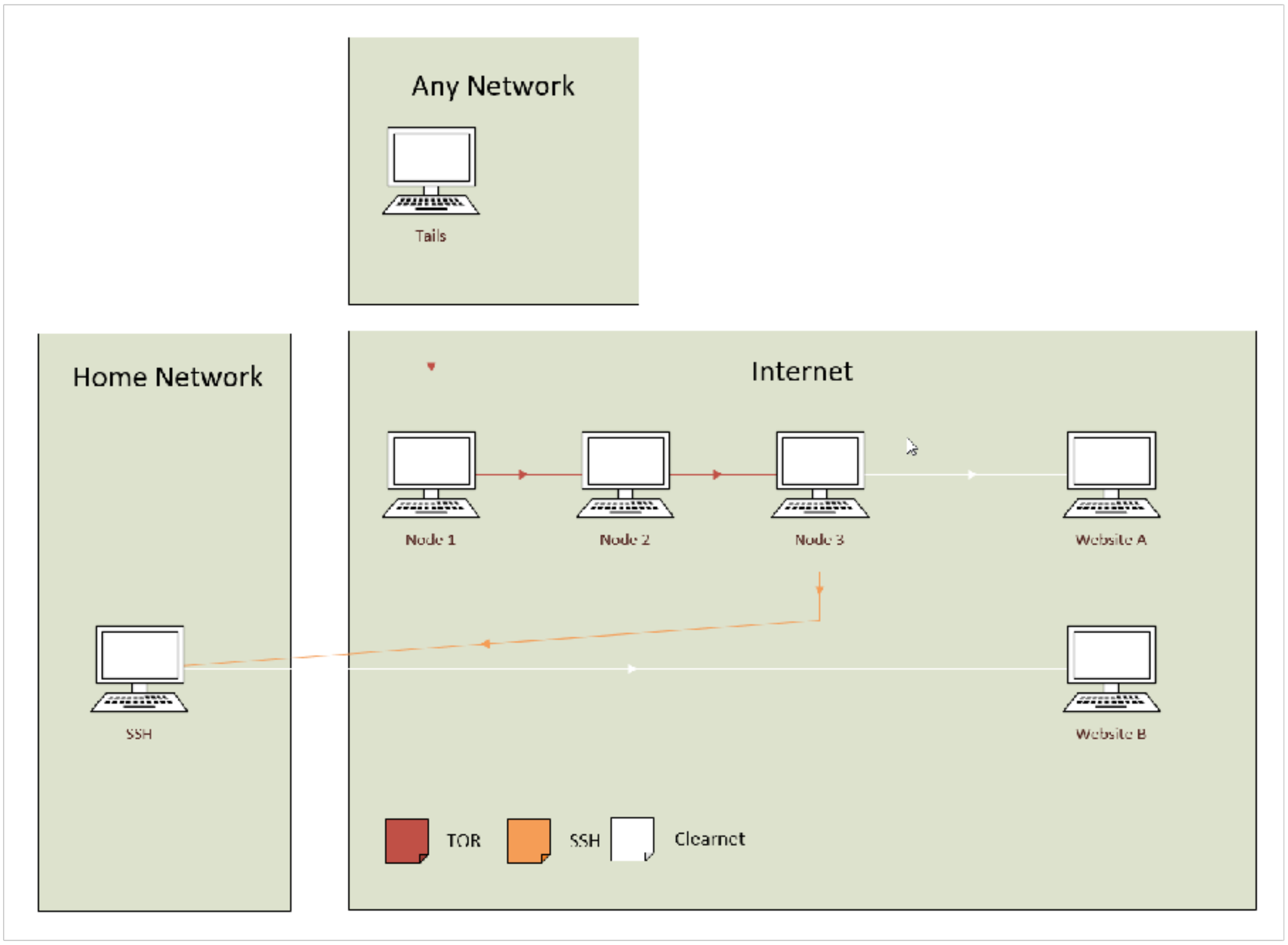


If you are using this “not so perfect scenario”, you can do the following:

- Use a Website like Google or many others that normally would block your connection. To be reminded again ! The complete Internet traffic (Data) that you send via my script through the Internet (from the SSH-server at your home to Website B for example), can be tracked and analyzed by your ISP, because it would be coming from a regular computer inside your home network ! Only the websites that you are visiting with the TOR Browser over the Onion-network are secure to visit without to being tracked (Website A for example). Of course, you could also use the “Unsafe Browser” of tails to visit any website you wish, as long the *Unsafe Browser” is enabled on startup of tails.

2.0 Using your own SSH-server that can be used when you are not at home

Scenario 02



If you would like to connect to the home ssh server externally from the Internet with Tails, there is some additional work to do.

- Port Forwarding of TCP port 22 (or any other desired port you would like to use) to the destination IP inside the home network needs to be enabled. This has to be done inside your router or firewall, depending what device you are using to connect to the Internet. Most users own a router for connecting to the Internet.

- Do not allow Root Logins over SSH (PermitRootLogin no)

- Allow only a single user to login over SSH (AllowUsers username)

- Enable a DYNDNS name for your WAN IP, because most ISP‘s don’t provide fixed IP-Addresses for the customers WAN interface. If you need a DYDNS name for your connections to the home network, I would call it a security hole that your often not aware of.

- As soon this system is reachable from the Internet, special considerations need to be applied for the security of our SSH daemon at home.

    - You may not use the standard TCP port 22. A very good replacement port would be TCP port 443 or 53.

    - Use a key instead of a password.

    - Disable all password logins after successful logins with a key.

    - Put the SSH-daemon on a schedule,if you know you’ll want it before hand.

    - Fake the login message to mislead the snoopers or any disliked person.

    - Only use SSH-V2, The older protocol V1 shouldn’t be used anymore.

    - Disable empty passwords for all accounts.

    - Always update your system to the latest versions available because your computer could be contacted directly from the Internet.

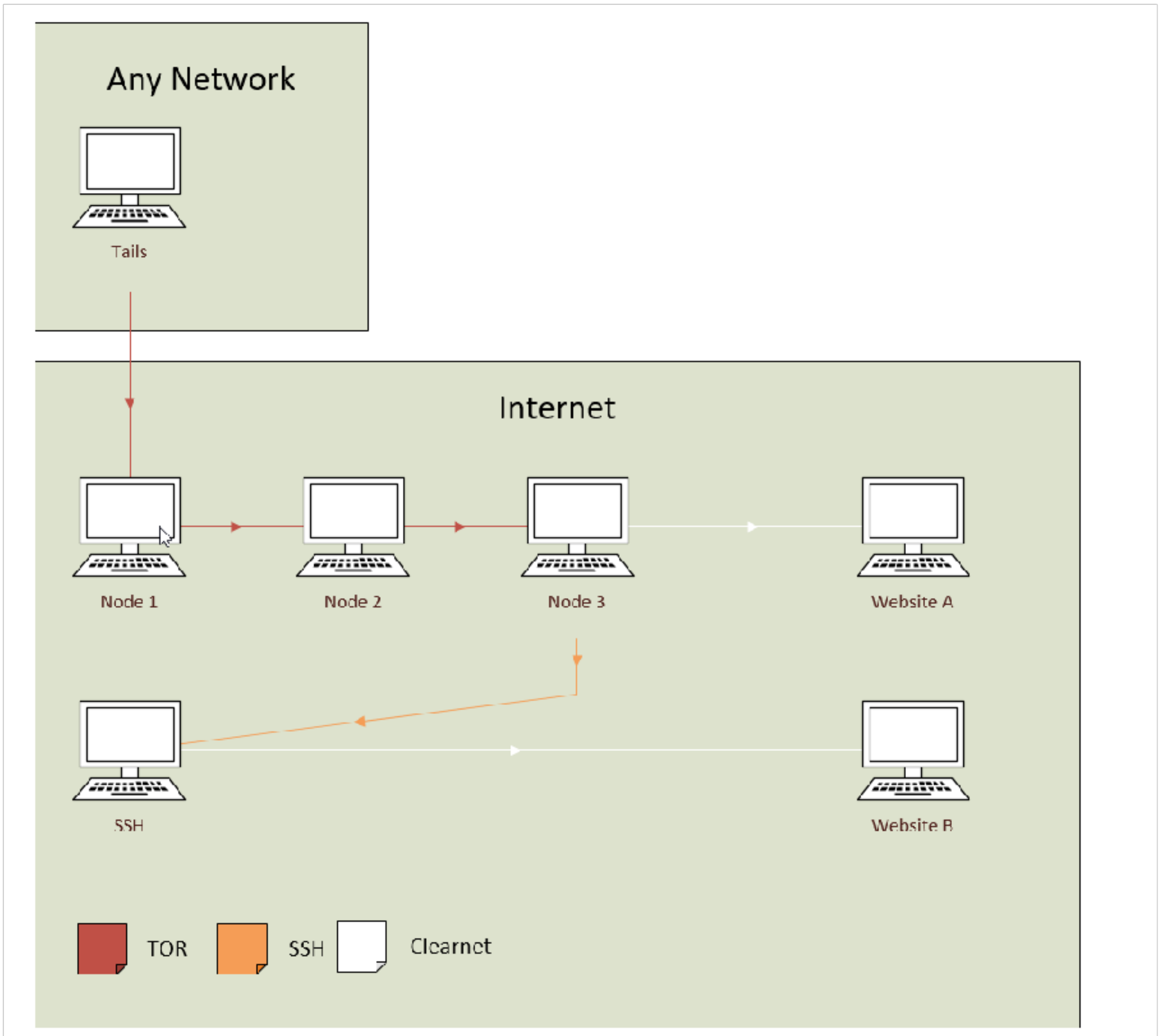
    - Your computer at home needs to be up and running all the time if you want to contact the your home server from any location via the Internet.

I would like to emphasize the importance of the fact, like in the previous scenario, all traffic that you send out over my script can be traced by your home ISP or a local government. If you are placing a server like this in a company environment, the staff of the involved company could also tracking all your traffic.

In this particular case it may even better to use the “Unsafe Browser” of Tails, of course it could also be tracked by the local ISP that your are currently connected to, but not by your home ISP if you are using the local “unsafe Browser” of Tails.

3.0 Using a remote SSH-server anywhere on the Internet (Best scenario)

Scenario 03 :



As you may now see, there are so many things that have to be configured correctly with SSH, especially if your own SSH-server can be reached from the Internet.
If you don't have a second computer inside your own network, the only suitable solution would be to find an SSH-provider anywhere on the Internet. This third option is the best option for all possible solutions to build your SSH-connection externally from Tails.

The difference from this Scenario compared to the two previous presented scenarios, is that your currently used ISP can only see the TOR traffic to the Internet. To be a bit more precise, your current ISP can only track the connection made to the first Node (Node 1) of the chain. Every TCP packet you send to through the Internet passes through 3 different nodes until it reaches the desired destination.

During this even the complete communication between the Node 3 (Exit-Node) and the foreign SSH server is encrypted until your packets have left the SSH-server.
With all the above described scenarios , one thing should highly emphasized.

You can’t hide the fact, from your currently connected ISP, that you are using the TOR-Browser or even Tails. If you really want to hide the fact that you are using TOR or Tails, you have to do that at your own risk and against the wise advice from the creators of Tails.

There are 2 possible ways to do this.

- **Start tails in “bridge-mode”**
  https://tails.boum.org/doc/first_steps/startup_options/bridge_mode/index.en.htm

Or

- **Use a ssh connection over TCP Port 443.This Port is used by default by websites with enccyption (https).**

So where should you start looking for a public SSH-server on the Internet ? As a good recommendation and starting point, you should have a look at the following URL :

https://shells.red-pill.eu/

Once again, we emphasize that you should only visit these multiple freeshell websites in the above link with the Tor-browser of Tails. If you are visiting them with a clearnet browser on linux or windows, you may have done a step to much to hide your personal information.

Another piece of advice from me is to use a fake-email address to register for a SSH-service of your choice. I’m sure you will find many SSH-providers on that ssh serverlist that meet some or all of your current requirements for a good SSH-provider.
And as a third and last piece of important advise from me, never create a username for a login that could be traced back to you.

- Some of them are free of charge, others are not.

- The process to create a valid account depends from server to server.
  - The only thing needed to create a account is a e-mail account.
  - A Email and a written postcard.
  - A little riddle to prove your knowledge about Linux and Unix in general.

- Some of them don’t ask for personal information about you, others would like to know almost everything about you.

- Not so many providers from that daily growing list, give a full shell-account including a new email address or the option for X11 forwarding.

- Many of them provide a little space to host a website on the server .

- A few of them have databases like mariadb or mysql.

- The provided disk space to store personal files is very often limited to between 20 mb or less.

- Some of these SSH-servers would work very well, as long you aren't trying to connect to them over the onion network. At the same moment you try to login over a public exit node from the onion network to that SSH-server, they terminate the connection immediately. Like the tor unfriendly websites we are already talked about, this servers are ware aware, that we are connecting over the ONION-network.

  But wait, Yes … there is a nice clever solution for this little handicap. first we make another SSH-connection to a other server that allows us to connect over onion-network. From our first SSH "Jumpserver" we make the desired connection to our second SSH-server.
  Not very easy to configure, but it will work.

Most shell-providers of this provided red-pill list do not allow the following "bad things".

- Only allows 1 active Connection with one registered login. Multiple connections with the same login would be detected and the user would be banned.

- Not allowed to "share" accounts with friends or other persons.

- The installation of your own software or malware files is strictly forbidden.
- The use of port scanners like nmap against other servers on the Internet is forbidden on most servers on the "Red-Pill" List.

- The use of software like "P2P" or "Torrent-clients" is strictly forbidden.

- On some servers is IRC allowed, on others completely forbidden.

- Some of these listed systems have hundreds of active users so be nice and keep In mind that there are other users as well and don't use up all the resources of a remote server. (Like CPU, Memory, Disk space or Bandwidth)

4.0 Preparations prior to use of this addon

To run this script from version 0.52 or higher you need the following things.

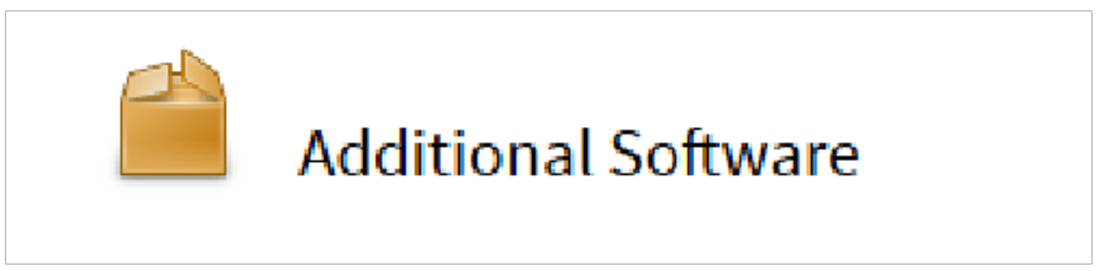
- One USB drive with at least 8 GB capacity.
- A current Tails version Version 4.14 or higher. Tails have an excellent documentation for the installation process itself.

  https://tails.boum.org/install/index.en.html

- A persistent volume within Tails with this 3 following mandatory options activated.

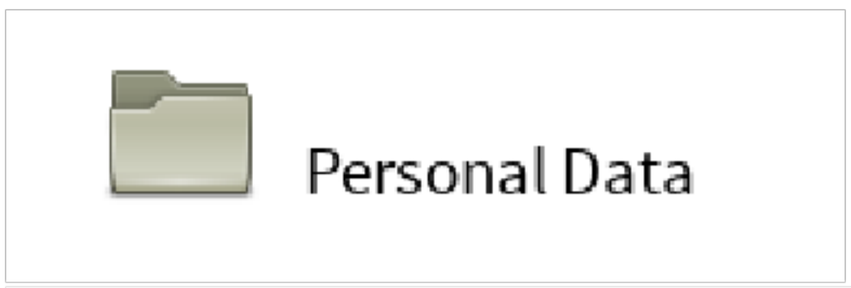
  https://tails.boum.org/doc/first_steps/persistence/configure/index.en.html

Without these 3 options enabled, the addon will not work correctly as expected. These 3 options are needed.



There are also 2 Debian packages (chromium and sshpass) that the addon needs to install once. Without them the script will not work correct.

Because all our connections inside the addon are created over SSH, we need this option to store our personal keys and the “known_hosts” files.



This one, is already active, as soon you create the persistent volume.

The following persistent volume options are not mandatory for the addition for Tails, but I would recommend to use them as well, I would say they are “nice to have optional features”.

I also like to store my personal bookmarks on the persistent volume.

If you are using multiple wifi networks to connect, this is may the perfect option for you.

This feature is a must-have option for all Tails users, that would like to make a real persistent Tails.

The remaining 6 persistent volume options available are :

- Printers
- Thunderbird
- GnuPG
- Bitcoin Client
- Pidgin
- Welcome Screen

If you use them or not, depends on your own personal choice depending on how you use Tails. My addon can also backup all the files of a persistent volume if you would like to do so.

At this stage, from now on, please remember to do the following when you start Tails.

- You have to open the persistent volume on every start of Tails if you want to use the addon. Of course, you can start Tails without the persistent volume activated, but the addon itself and all the data is stored on this persistent volume aren’t usable, even the stored wifi passwords aren’t usable.

- If you are using special foreign chars for the password of the persistent volume you have to set the correct keyboard layout first, otherwise you will be typing the password with the default english (USA) keyboard layout in the Tails greeting screen. Prior to version 4.12 it was not possible to store all settings from the greeting-screen.

- The administration password of Tails also needs to be set on every start of Tails if you don’t store your settings of greeting-screen . If you don’t set an administrator password, the addon won’t be able to change the default local firewall. The script changes only one unique little setting inside the firewall of Tails.

  The changed setting is

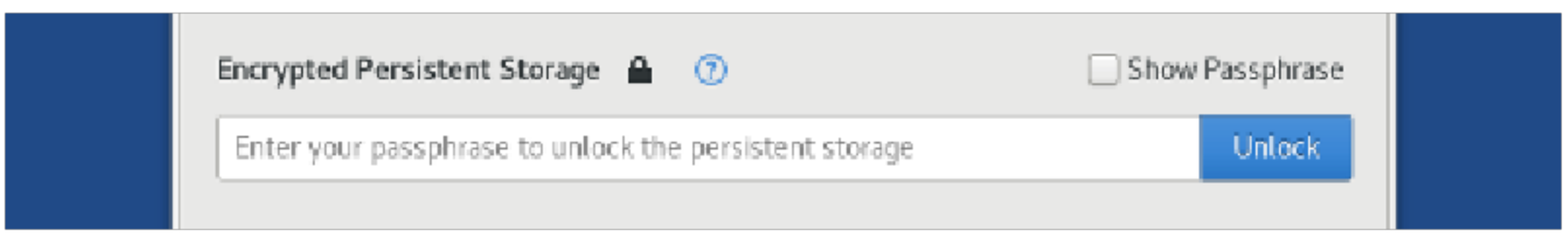
  sudo -S iptables -I OUTPUT -o lo -p tcp --dport 9999 -j ACCEPT

  This change allow us to build a local socks5 proxy over SSH. Without this modification, the predefined default rules from iptables would block any connection attempt made to port 9999 from the Loopback device 127.0.0.1

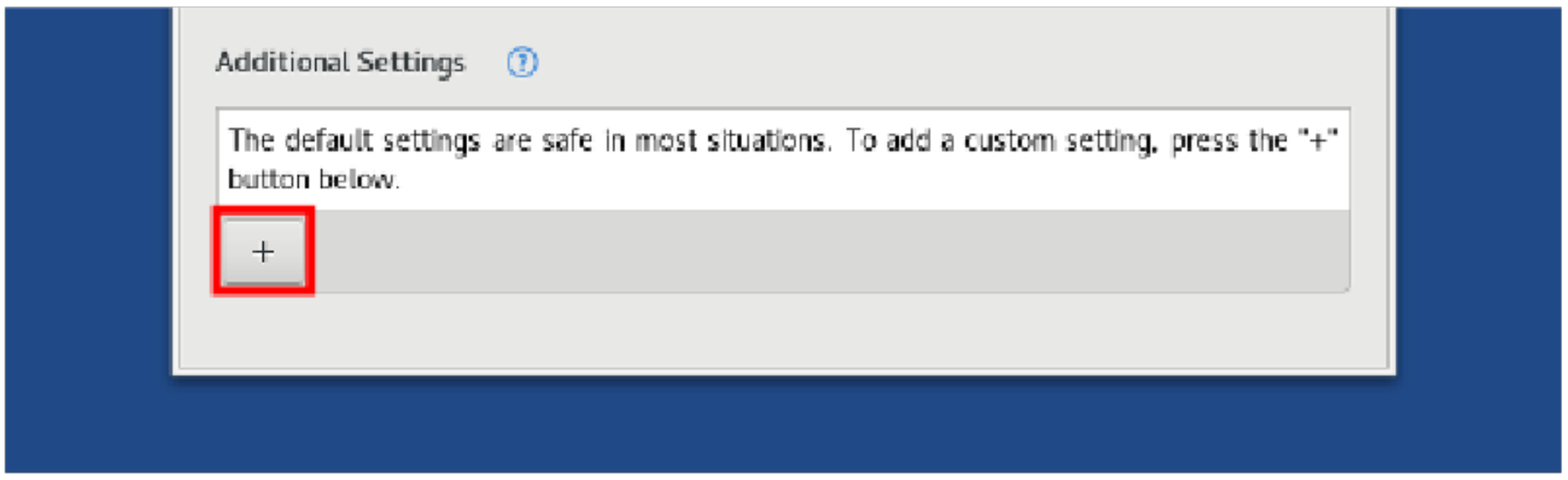
5.0 Installing the addon

After the creation of the persistent volume you have to reboot to make the persistent volume active. After the next startup you may change, the Language and the Keyboard layout:

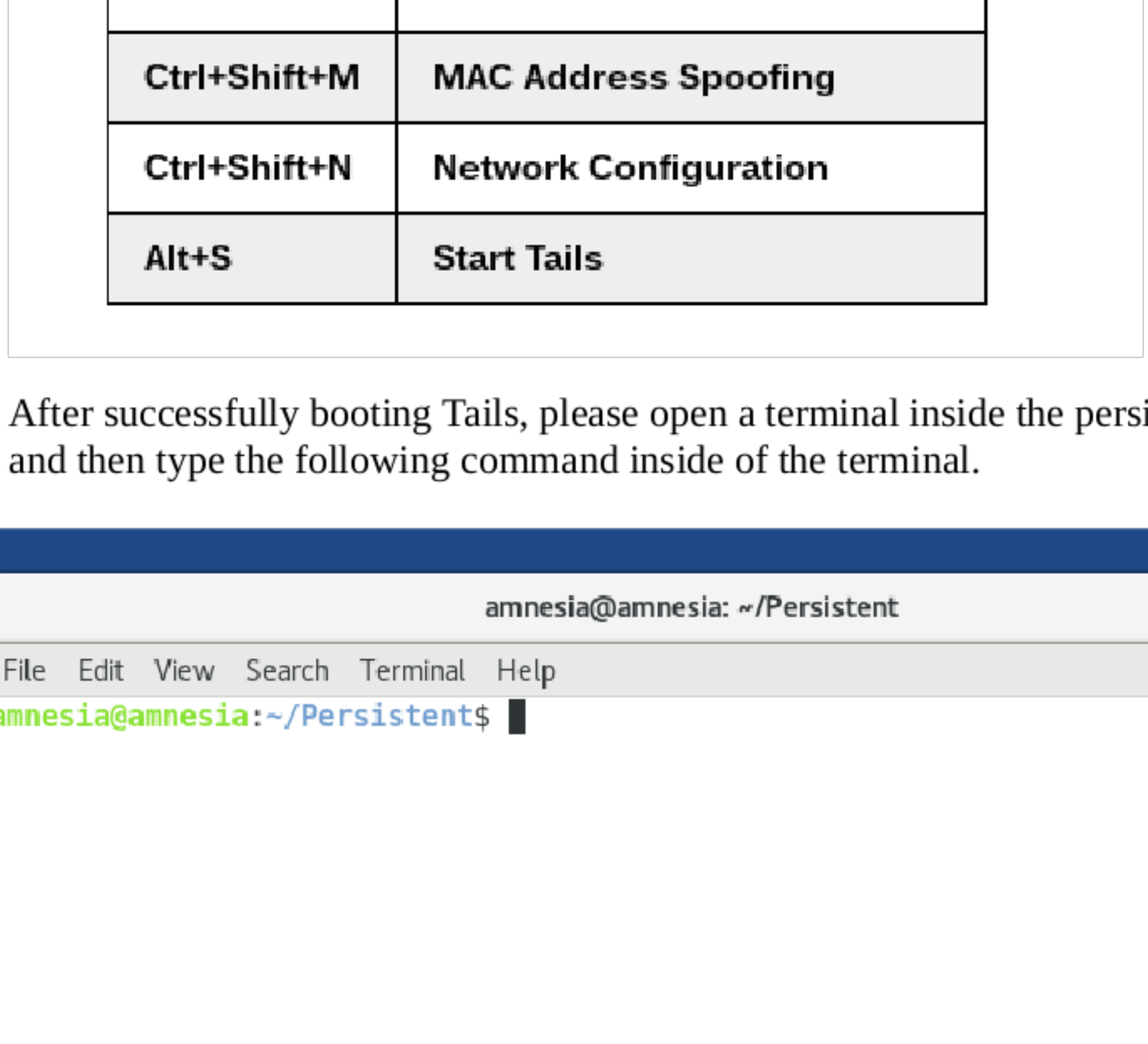
And then of course, you activate the persistent volume



And as a last option, you set a administration password.



A little tip with the keyboard shortcuts on the startup screen of tails:



Keyboard shortcuts

Alt+L	Language
Alt+K	Keyboard Layout
Alt+F	Formats
Alt+P	Encrypted Persistent Storage
Alt+A	Additional Settings
Ctrl+Shift+A	Administration Password
Ctrl+Shift+M	MAC Address Spoofing
Ctrl+Shift+N	Network Configuration
Alt+S	Start Tails

After successfully booting Tails, please open a terminal inside the persistent folder, and then type the following command inside of the terminal.

---

git clone https://github.com/swtor00/swtor-addon-to-tails

Depending on the speed of your current Internet connection, after a while you should find a folder called “swtor-addon-to-tails”. Don’t’ worry about the https connection to github (a microsoft company since June 2018). You are already using the onion-network to connect to the github-server and you are not leaving any traces or evidence on the github website or that you are using this script or even Tails.

Before you do anything inside the directories of the addon itself, you should first make a few important decisions about the use of the addon itself. The configuration of my addon is written in a single file. It can be edited over the Gnome Editor (gedit) or even vim / nano if you would like to edit the file this way.
Prior to executing anything, the complete path to this configuration file is as follows.

~/Persistent/swtor-addon-to-tails/swtorcfg/swtor.cfg

The default swtor.cfg configuration file for version 0.39, looks like this.

```
VERSION:0.39
STATE: BETA
HOMEPAGE: https://github.com/swtor00/swtor-addon-to-tails
JOTV :
-----------------------------------------------------------------
"Programming today is a race between software engineers striving to
build bigger and better idiot-proof programs, and the Universe
trying to produce bigger and better idiots.
So far, the Universe is winning."
-----------------------------------------------------------------
OPTIONS FOR THE SWTOR-ADDON

IMPORT-BOOKMARKS:NO
GUI-LINKS:YES
BROWSER-SOCKS5:YES
CHECK-UPDATE:NO
BACKUP-FIXED-PROFILE:NO
BACKUP-APT-LIST:NO
```

In the last few lines, you see all the relevant entries for my addon. All addon options are in CAPITAL letters.

If you would to check for an update on every startup, you have to replace

CHECK-UPDATE:NO with the entry CHECK-UPDATE:YES

After this little change, my addon contacts the github server on every startup to check if there is a new version to download. If you would like to manually make this update check for yourself, open a terminal and type the following.

```
cd ~/Persistent/swtor-addon-to-tails/scripts
./update.sh
```

The script will inform you if there is a new version.

Warning :

All local changes made to all files including the main-configuration file swtor.cfg are overwritten with the default values stored on the github server. You have to re-apply all the changes you made again ! Of course only, if you made some changes to the configuration file swtor.cfg.

If you would like to use the current script version without the tracking of git software, you could execute the following terminal commands to accomplish this task.

```
cd ~/Persistent/swtor-addon-to-tails
rm -rf .git
rm ~/Persistent/scripts/update.sh
```

At the same time you remove the .git directory from the addon directory, you cannot execute ./update.sh again. Therefore it should be written in the configuration file swtor.cfg

CHECK-UPDATE:NO

What are the other possible settings inside this configuration file ?

IMPORT-BOOKMARKS:NO

If you change the value to YES, predefined swtor bookmarks are directly imported on the first startup of the addon. The default value is NO, for one big reason.
I don’t like to hear, that someone's large personal bookmarks are accidently overwritten by my script on first startup.

GUI-LINKS:YES

If you change the value to NO you can only start the script over terminal. Most users should use the default value YES.

BROWSER-SOCKS5:YES

Currenty it should always be YES , In the near future it may come with other settings like BROWSER-PROXY or RDP-CONNECTION / VNC-CONNECTION.

BACKUP-FIXED-PROFILE:NO
BACKUP-APT-LIST:NO

If you activate both options to the value YES. What does this exactly mean ?

In the case you would like to backup the complete persistent volume, the size of the backup will be around 200 mb or even more. If you leave the values at the default state NO, the created backup will have the size somewhere between 3 - 5 mb.

6.0 Configuring the required SSH-connection for the addon

For this next configuration step, we need a valid SSH account from anywhere on the Internet or our home-server. You may even try first to test this SSH connection with another operating system like Windows, Linux or even an Apple system which contains all the software needed to establish a SSH connection, generally all modern operating system … (including Windows 10 Version 1803 or higher) do have this SSH-software included.

In my opinion, DO NOT even think about using this. Under all circumstances, it's an insanely bad idea to use this SSH-connection anywhere outside of the Tails-system !! If you really want to use a socks5 SSH connection to hide your browser traffic with any other operating system than Tails, you should create a complete new login on a remote SSH-host only for that purpose ! If you use these SSH-credentials outside of Tails, you would leak your current WAN IP-Address to the owner of the SSH-host immediately the moment you try to connect over SSH without the protection of the Onion-Network in the background. A simple command "who | more " inside a terminal would list all current connected users and the corresponding IP-Address from the location a user is connected. An example of the command could look this.

```
eao         pts/7       2018-xx-23 17:27 (85.220.101.10)
dyama       pts/8       2018-xx-25 02:07 (158.3.77.185)
tt0077      pts/9       2018-xx-25 20:08 (57.41.129.24)
jose1711    pts/11      2018-xx-08 09:14 (185.177.234.117:S.0)
piny         pts/13     2018-xx-27 07:48 (237.47.33.159)
```

In the case that we are using the Onion-Network inside Tails, the printed IP-Address would only be from our used Exit Node 3.

An SSH account normally consists of the following information for a successful connection.

- A username like digit1 including a valid password
- A destination port. The default port for a SSH communication is TCP port 22.
- A valid DNS-Name or a IP V4 IP-Address

All the configuration files for an SSH connection reside inside the directory /home/amnesia/.ssh. If this directory is empty, it means that we have never contacted any SSH-server before with our current Tails system.

To test our connection and see if we can successfully login over SSH, we need to open a terminal and execute the following command. (You can replace the values in this example with the values provided by your own SSH-provider.)

ssh -p 22 digit1@10.0.1.66

Description of the above command in detail :

ssh	The command to communicate encrypted with the other host.
-p 22	22 is the default port for a SSH connection. Due to the behavior of SSH , it isn't always necessary to add -p 22 for every connection. If your SSH-providers doesn't use port 22, then the -p option should be used every time.
digit1	The username of the SSH login
10.0.1.66	The IP-Address of the remote server. We could also provide a DNS-Name instead of a IP-Address.

In the case of the following scenario where we have never made an SSH connection with our new Tails Medium to an SSH-server with the IP-Address 10.0.1.66, we will see a warning like following one.

The authenticity of host '10.0.1.66 (10.0.1.66)' can't be established. RSA key fingerprint is 90:8c:7d:f8:ae:1a:09:60:44:08:3b:d9:c9:f7:c4:76.
Are you sure you want to continue connecting (yes/no)?

This is the so called "public fingerprint" of the SSH-Server we are trying to connect to. The moment we type "yes" inside the terminal, the public fingerprint for this specific server 10.0.1.66 is then stored inside the file ~./ssh/known_hosts.

After storing this public key inside Tails, on every connection we make to 10.0.1.66, the already stored value inside the file /ssh/known_hosts will be compared against the value that the server provides upon connecting. If there is a match of both values, we can continue to establish our connection. If the two values don't match, then there is something wrong ! If you find the following entries inside your log file, be very carefully with your next action.

WARNING : REMOTE HOST IDENTIFICATION HAS CHANGED. IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY !

- It may be that someone is trying to make you think that you are connecting to the host 10.0.1.66 ,but you aren't, this is called a "man in the middle attack".

  https://en.wikipedia.org/wiki/Man-in-the-middle_attack

  Someone could now steal your current password for that particular SSH-host or even worst, steal your current public SSH key if you ignore this warning and connect to this possible evil or nasty SSH-Server !!!!

- Or that the public key of the server has been replaced for some reason, possibly the remote server SSH-Server was replaced due to a hardware failure.

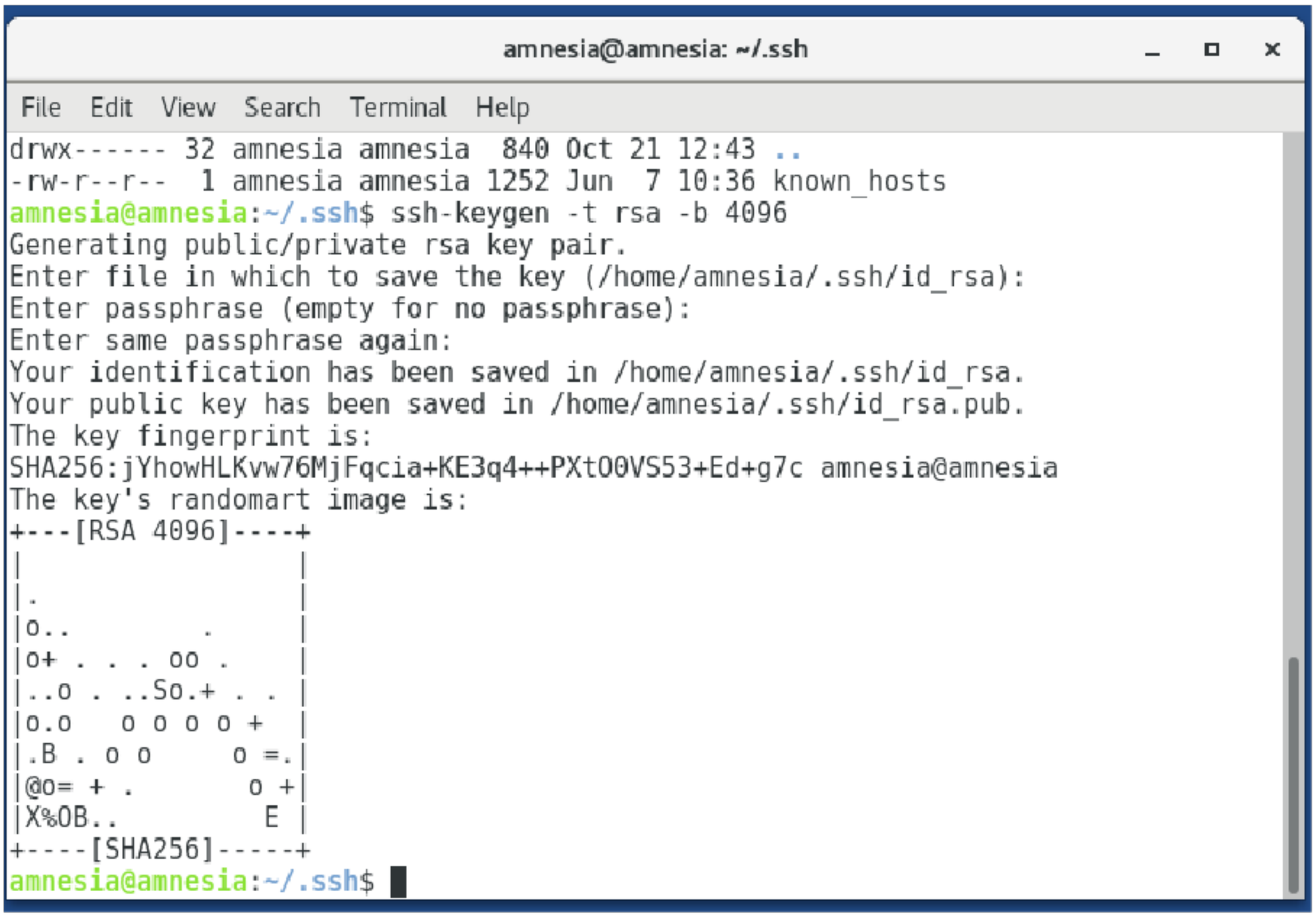
Next, you should see the password login, for the requested userlogin on the remote server. You can now type the password for that account and the remote shell should appear straight afterwards.
Provided we can login to the remote server without any error, our next step would be to make this login password free in the future. We can close the remote shell on the remote server by typing "exit" or using the keyboard shortcut <ctrl><d>. Of course you could also customize a few things like changing the current password or a few other things also. If you change the current password, please write it down somewhere and store it in a safe place. If you have to start with a new empty Tails (Clean Tails Clone) , you may need the change password, to transfer the backups you made.

Our next terminal command is used to create the private / public key pair for all future SSH-communication inside Tails. The command to accomplish this, is the following one :

---

ssh-keygen -t rsa -b 4096

After a short initialization time to generate these public and private keys , we have an output like the following one. This is now our own personal "holy-grail" of encrypted communication for use inside Tails and should be saved on a regular basis.



Now we can copy our public key to our SSH-server. There is a special SSH command to do this.

---

ssh-copy-id -i ~/.ssh/id_rsa.pub digit1@10.0.1.66

Some older Unix systems don’t support the ssh-copy-id program, so with a few little bash-tricks it’s possible to transfer the public key to the foreign SSH-server with the standard Unix commands every system should clearly understand.

---

cat ~/.ssh/*.pub | ssh digit1@10.0.1.66 'umask 077; \
cat >>.ssh/authorized_keys’

By now it should be possible, to make an SSH connection with Tails to the SSH-system 10.0.1.66 without any password or any other additional typing with the keyboard.
I recommend the use of at least 2-3 SSH-servers inside of the swtor-addon .

- For every ssh-host you would like to connect to, you have to execute the command ssh-copy-id or you have to type the password again on every connection you make !

- You should test every additional ssh-connection carefully so that there is no confirmation needed like adding the public key to the known_hosts file inside the directory ~/.ssh.

As soon as you have at least one SSH connection that works properly, you can create the configuration file that is needed by the addon. Inside the “doc” directory of the addon , you will find a small example pdf (sample-configuration.pdf) that explains exactly how to create this configuration file. The configuration of all possible SSH connections that this addon can use are defined in this single file.

~/Persistent/swtor-addon-to-tails/swtorcfg/swtorssh.cfg

SSH itself is a very complex piece of software, if you would like to have more information about SSH in general or you need some cool advanced troubleshooting tips, you should have a closer look using the TOR-Browser to navigate to the following url’s.

https://www.allitebooks.in/ssh-secure-shell-2nd-edition/

https://www.openssh.com/manual.html

Tips for troubleshooting :

All SSH-connections made with the swtor addon have a “verbose” output on all actions.
You find all the logs inside the following directory.

~/Persistent/swtor-addon-to-tails/swtorcfg/log

7.0 Executing the addon the first time

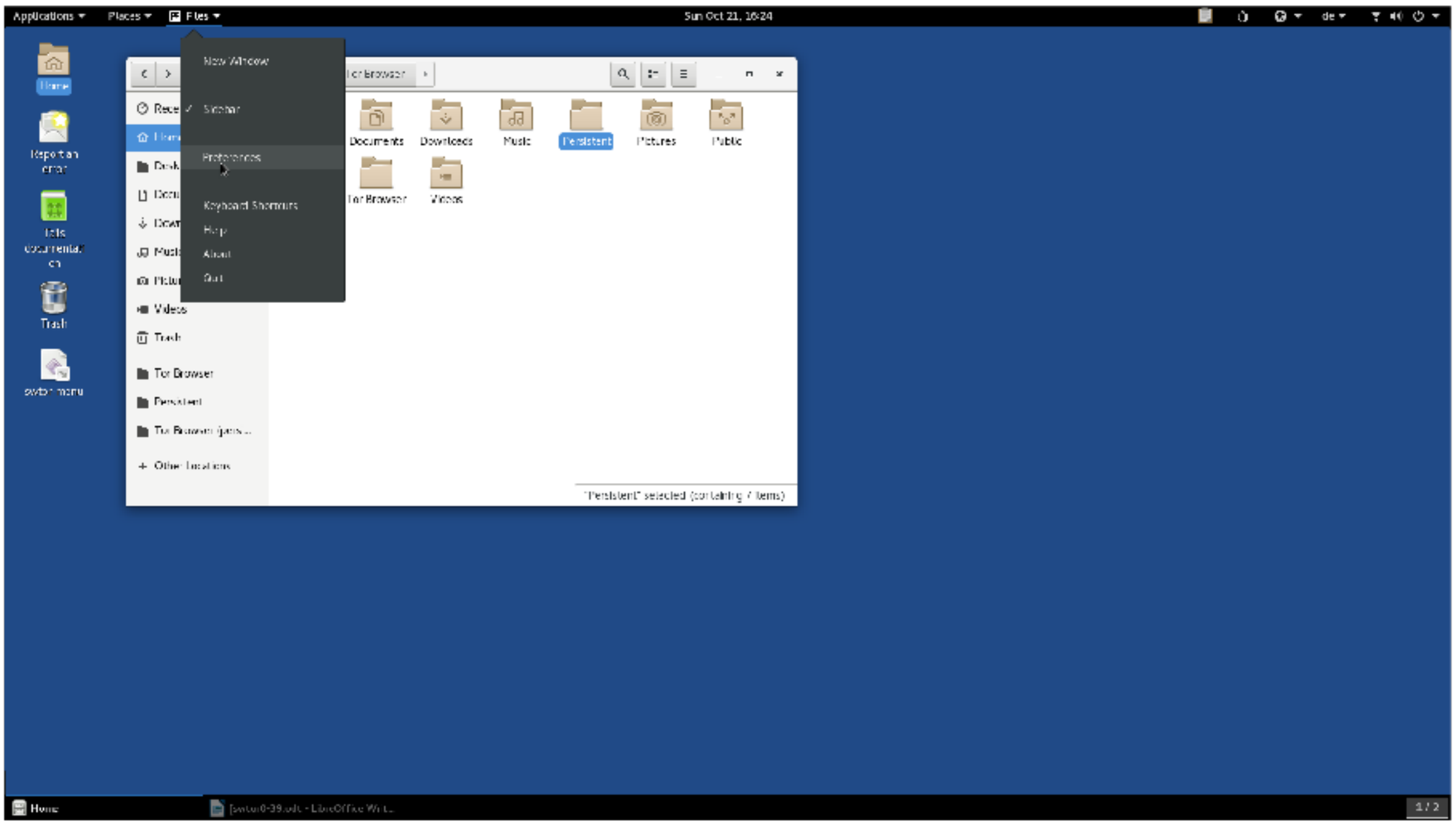
The default behavior of the Tails-System doesn't allow the execution of shell-scripts over the GUI Interface. If you don’t want to execute the scripts over the GUI of Tails, you must execute the shell-scripts with a terminal. In this particular case you can skip the configuration of the GUI to execute shell-scripts and write directly ./swtor.-setup.sh to start the script.

```
cd ~/Persistent/swtor-addon-to-tails/scripts
./swtor-setup.sh
```

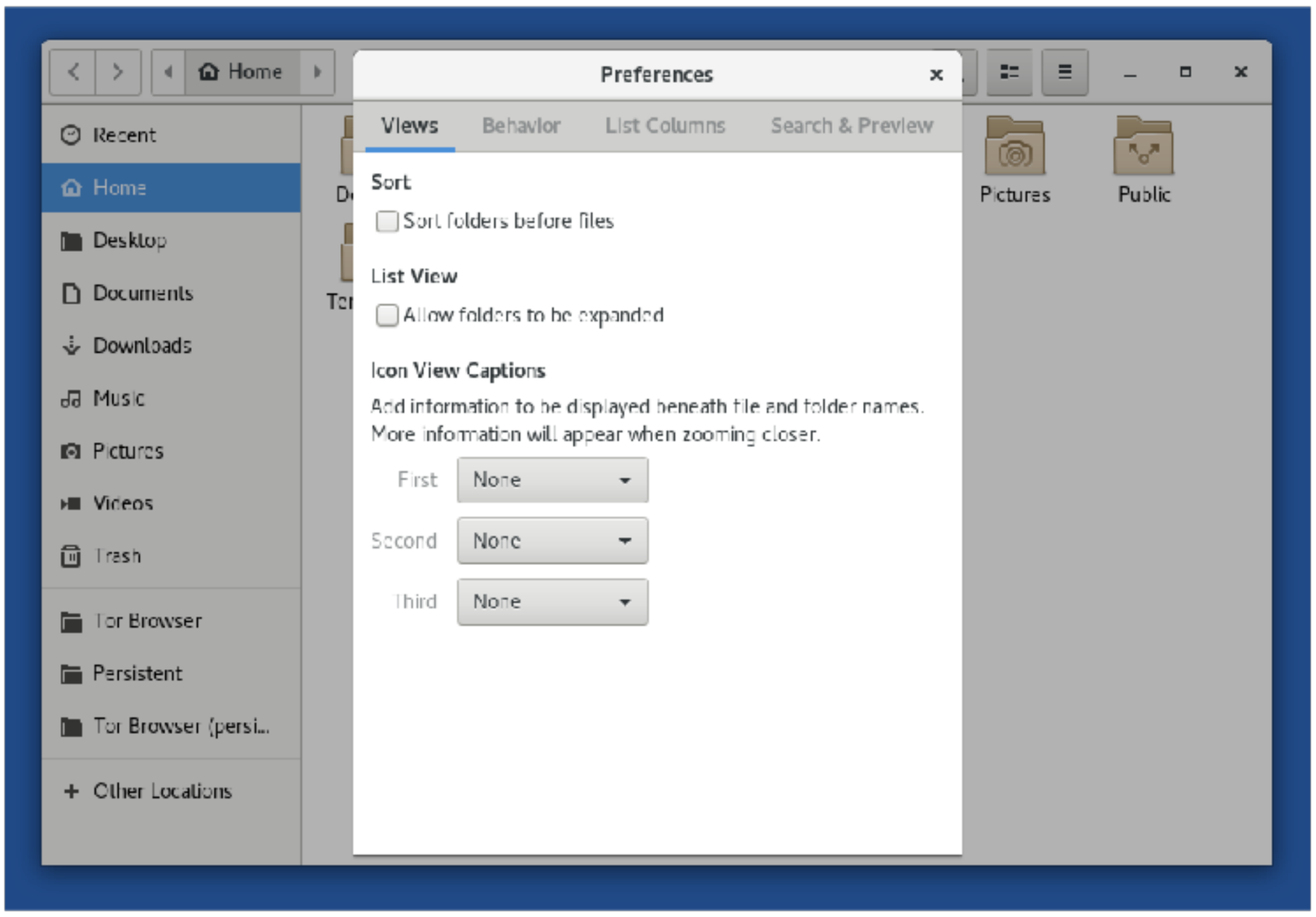
Please make the following little changes inside Tails to make all shell scripts generally executable. You have to repeat this step, on every startup of tails !

1.) Open “Home” on the tails desktop

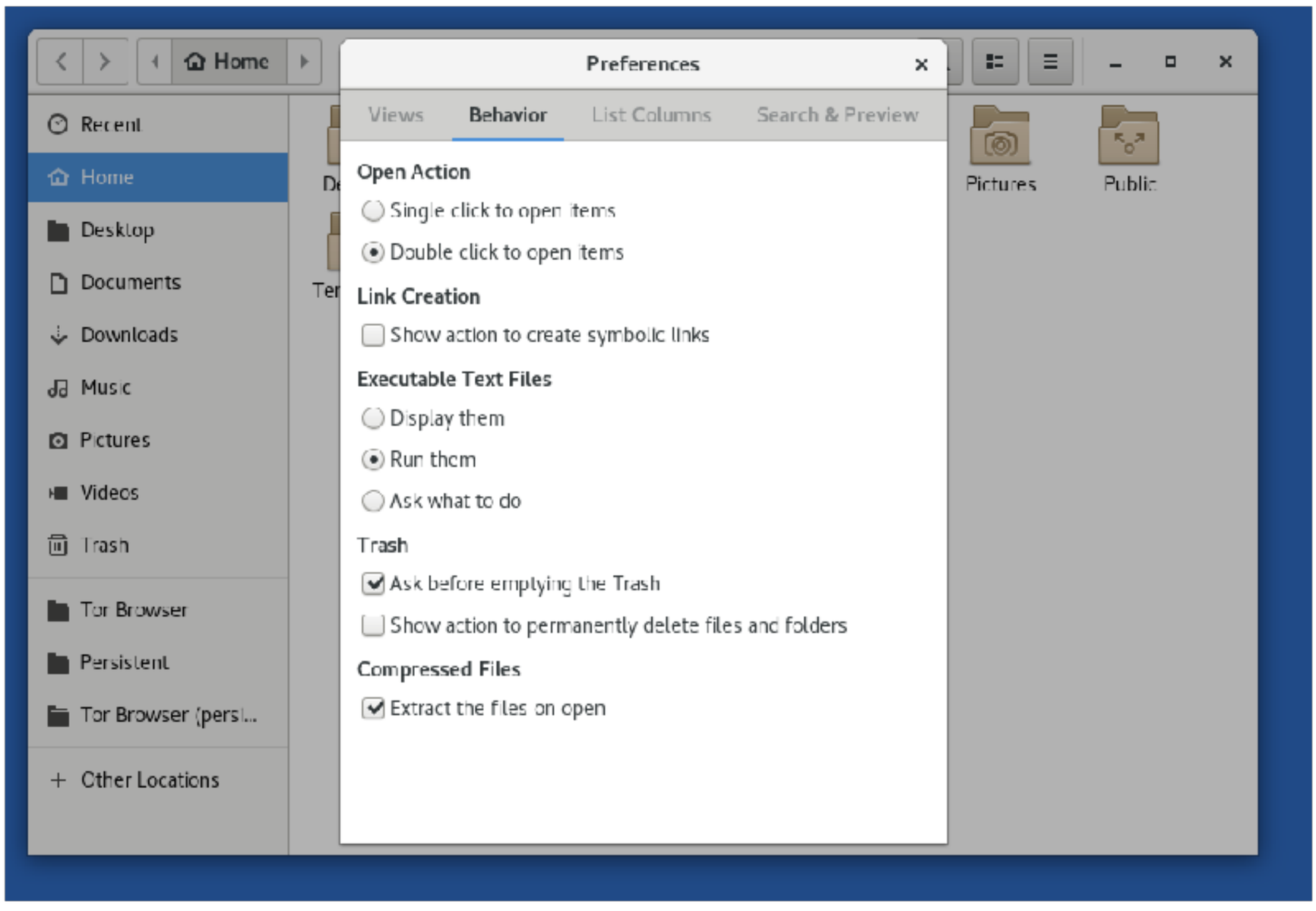
2.) Open “Files” and “Preferences”



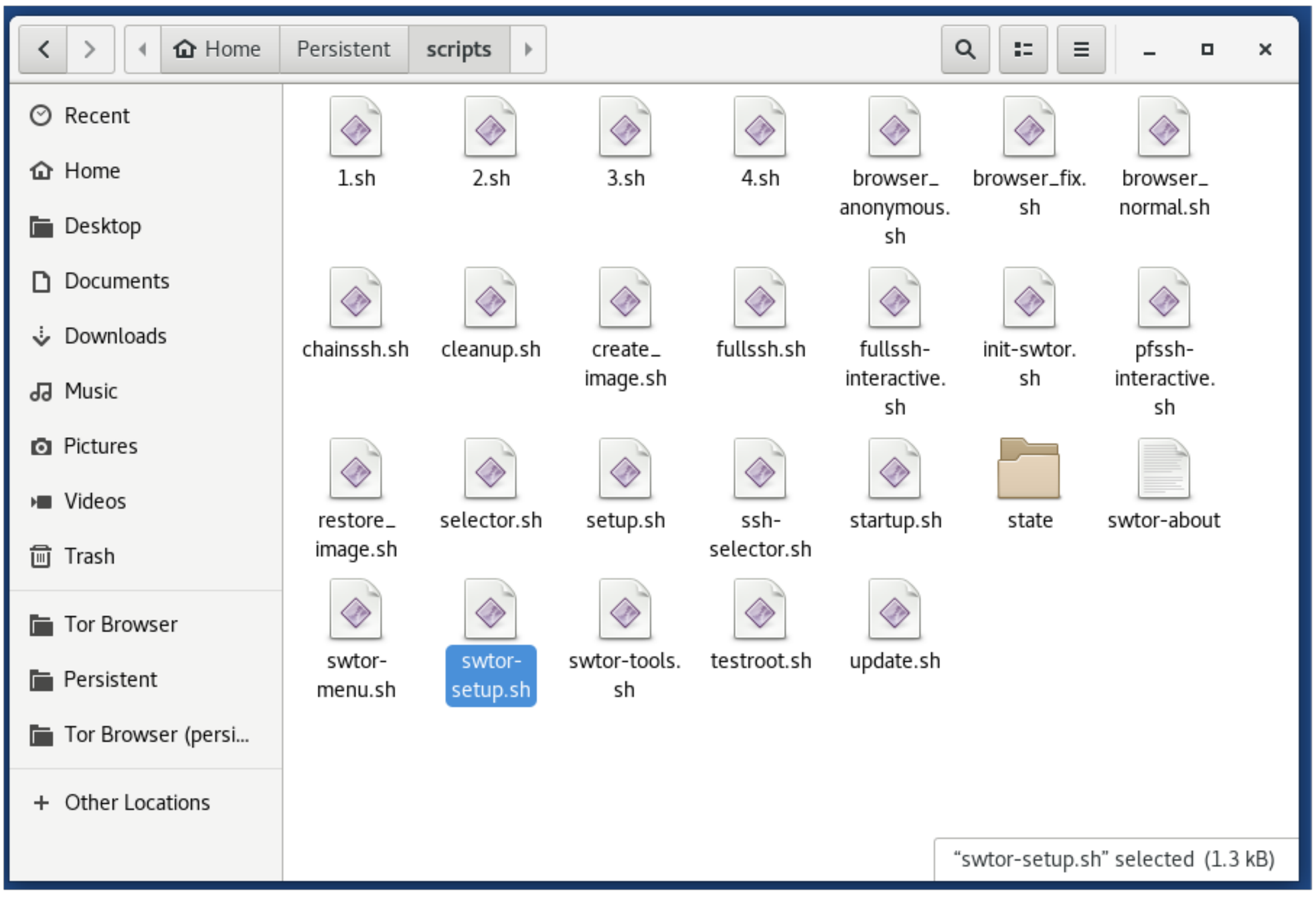
3.) Open the “Behavior” Tab



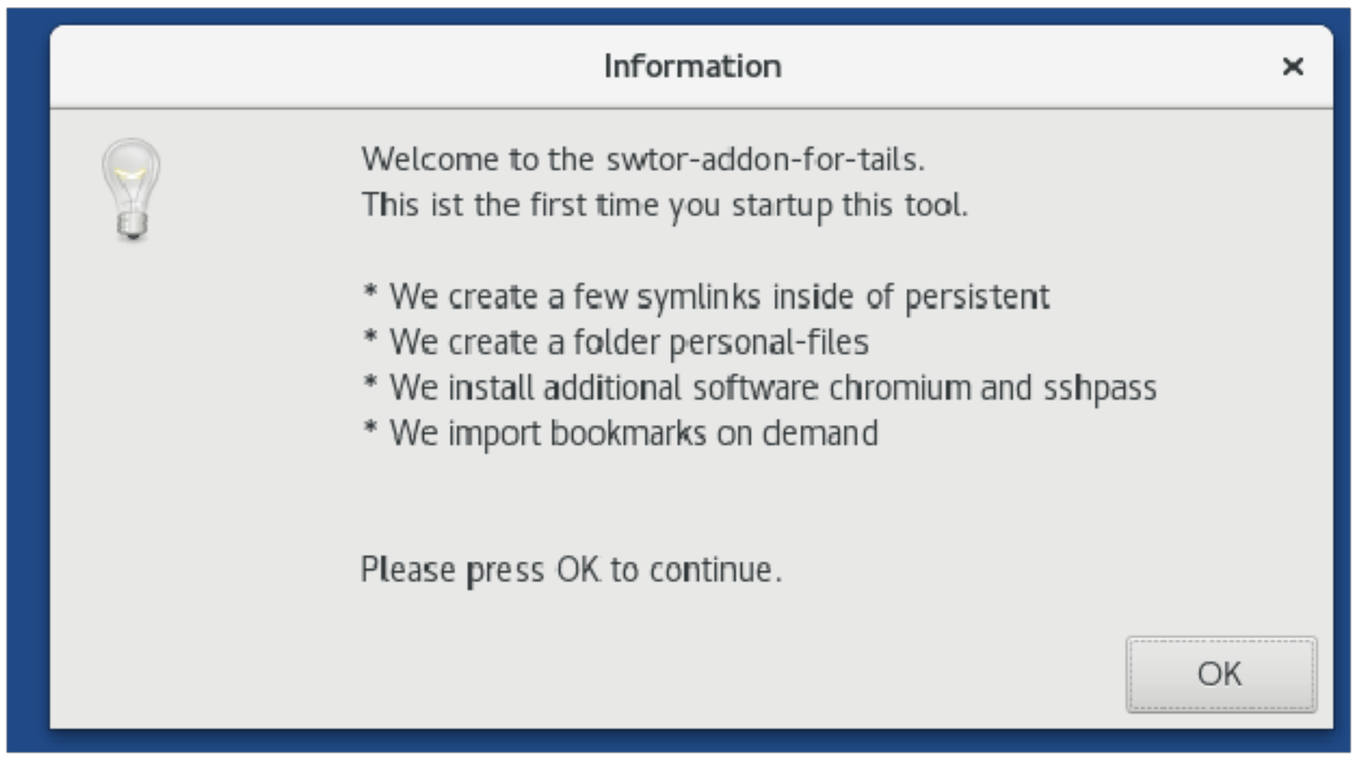
4.0 Activate the option “Run them”



Next, navigate to the folder ~/Persistent/swtor-addon-to-tails/scripts



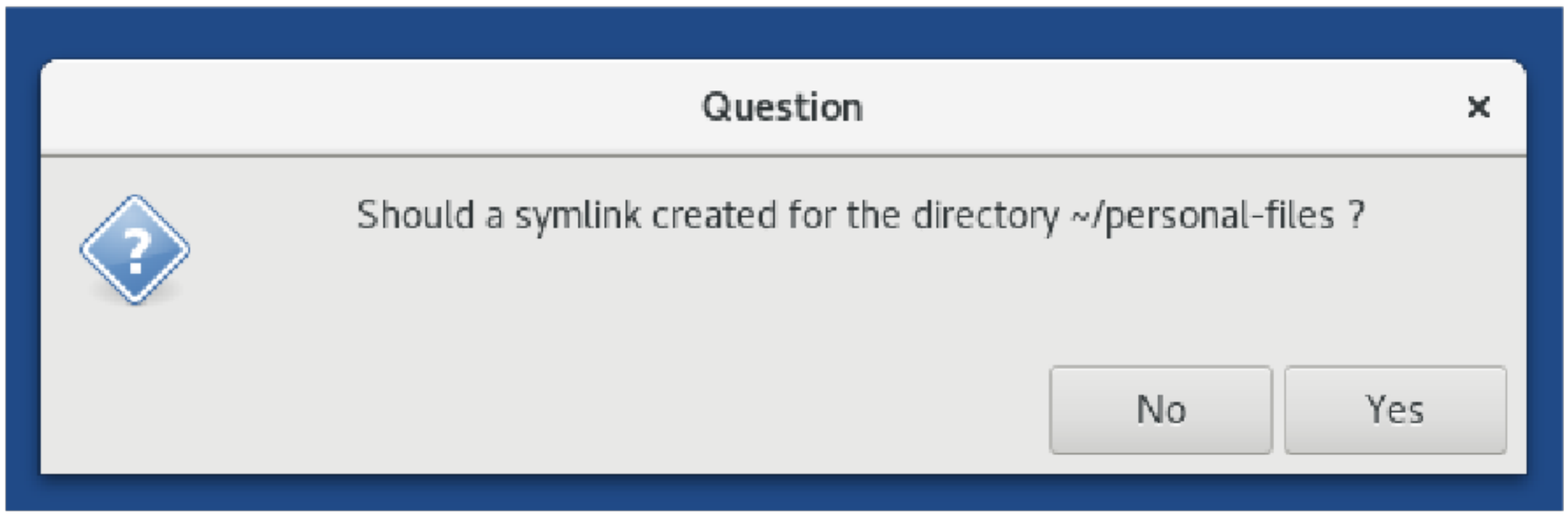
Please execute the file "swtor-setup.sh" with a right mouse double click → execute or do a double-click on it. Within a few seconds, you should see the first window pop up.



The following 4 symbolic links are created during this first startup

~/Persistent/scripts	**/home/amnesia/Persistent/swtor-addon-to-tails/scripts**
~/Persitent/settings	**/home/amnesia/Persistent/swtor-addon-to-tails/settings**
~/Persitent/swtorcfg	**/home/amnesia/Persistent/swtor-addon-to-tails/swtorcfg**
~/Persitent/doc	**/home/amnesia/Persistent/swtor-addon-to-tails/doc**

The next question, is about a symblic link to the personal-files directory.

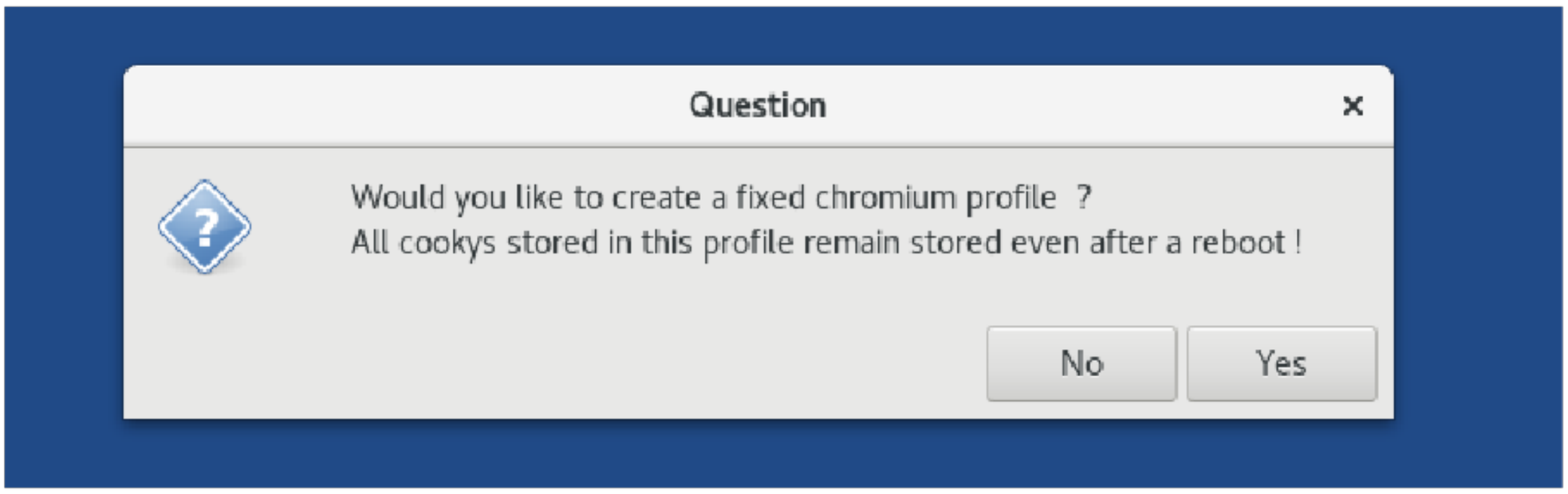


The following single directory and a optional symbolic link (see above)

~/Persistent/personal-files

is created.

The next question is about the use of a “fixed chromium profile”. By default the swtor-addon-use 2 predefined profiles for surfing. If you decide to create a fixed profile, there will be a additional directory inside of the personal-files.



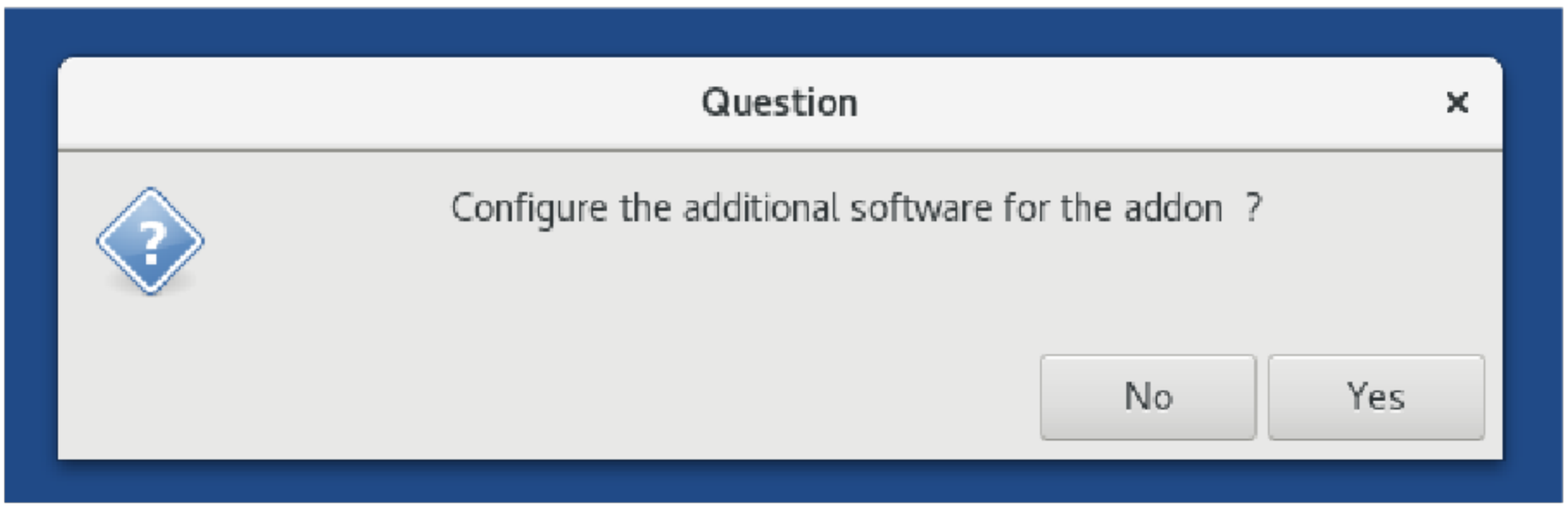
If you use a fixed profile inside of ~/personal-files/3, then it is possible to use

- Bookmarks and Cookys inside of the fixed profile that remain after a reboot.

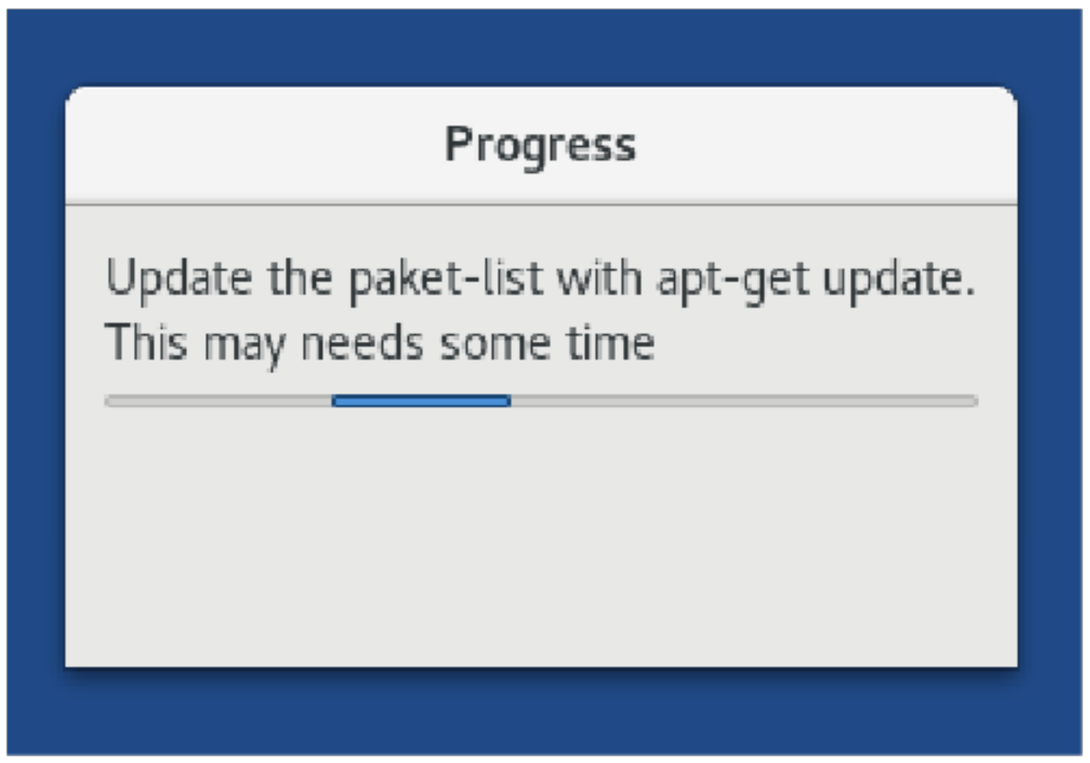
Warning :
The 2 default chromium profiles inside of ~/Persistent/settings are completely removed on every SSH-connection change ! Do not try to save Bookmarks on this 2 profiles !

If your current Tails already contains the two following Debian packages in the Persistent additional software feature, you could skip this part of installation.

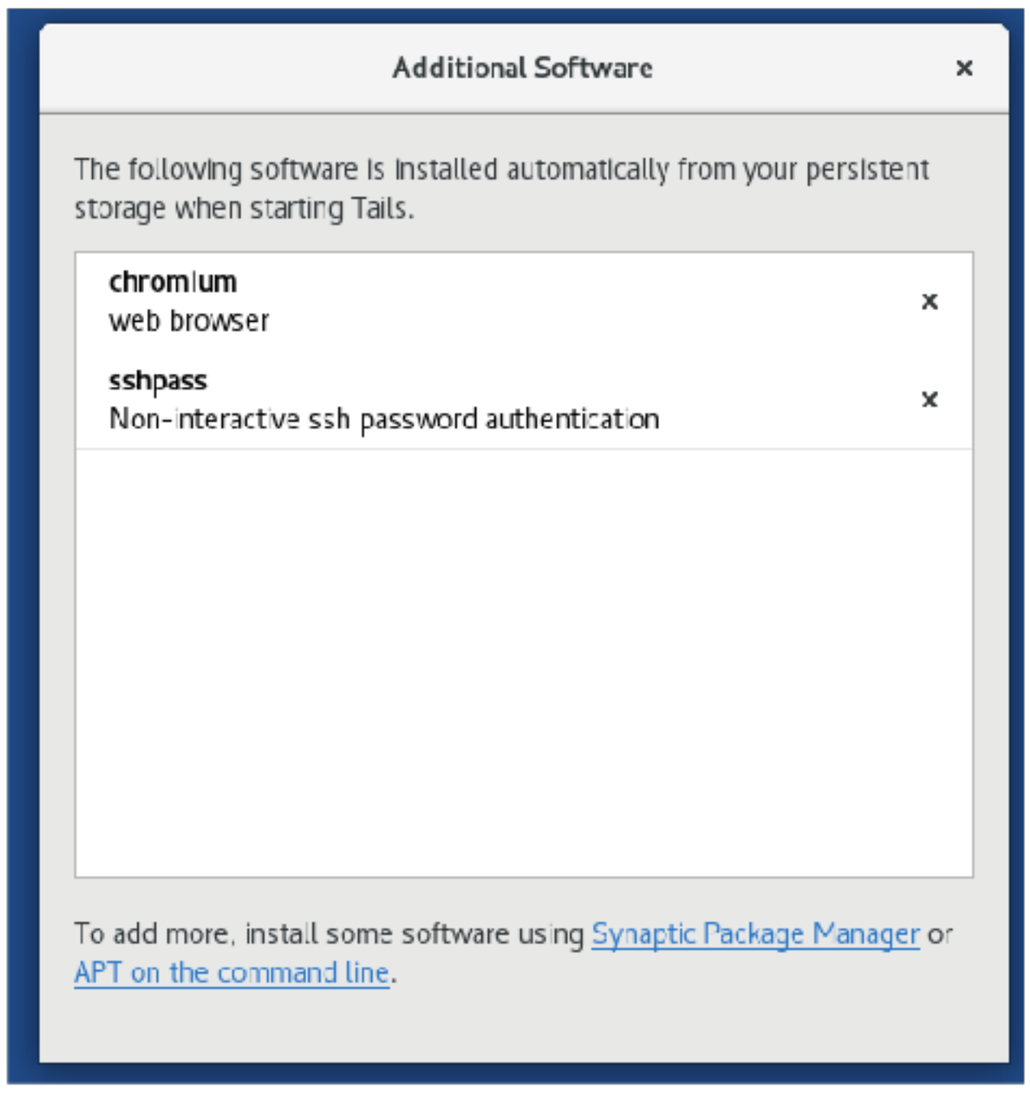
- chromium
- sshpass



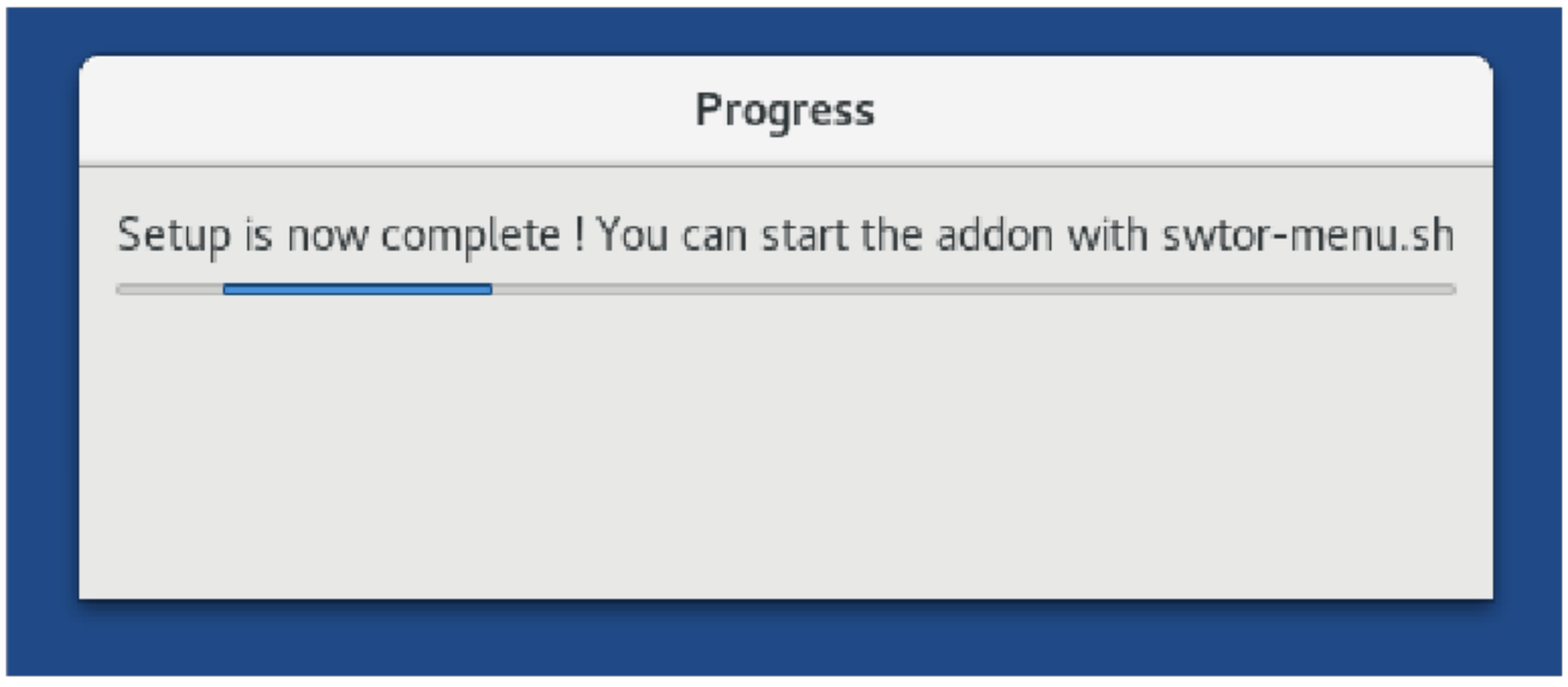
In the case the answer is yes, we do install the two Debian packages on the first startup of the swtor-setup.sh script, to install this additional software we will have to type the administration password.



The startup of apt-.get update may take longer than normal because we are installing these two packages from the official Debian archives. After the installation of the software, the Additional Software Configuration in Tails should look like the following picture. This installation has only to be made once. As soon Tails is started in the future with the encrypted persistent volume, this listed software will be installed without any user intervention.



The last screen inside of swtor-setup.sh should be this following one.



Little reminder :

Because we have now executed this script, we have a few shortcuts to the most important folders of the addon itself.

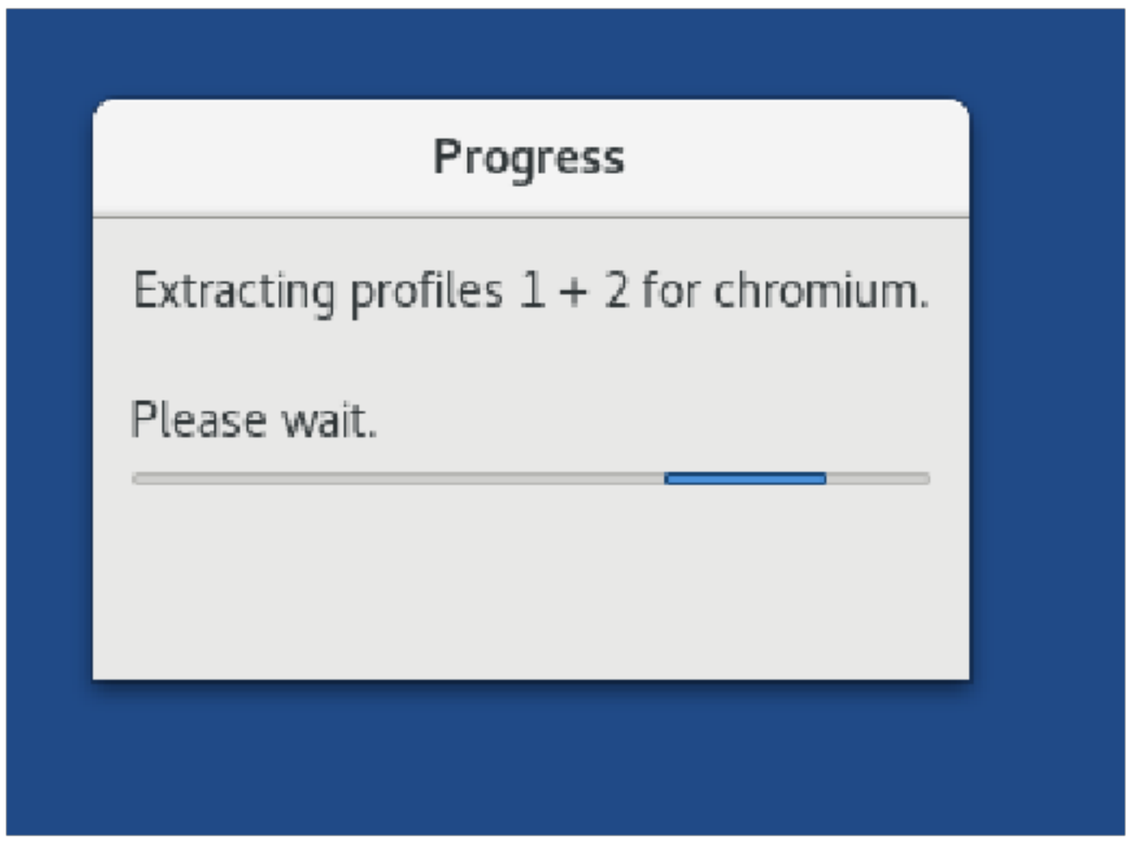
8.0 Use of the addon, after the first initial run

As next we do start the addon itself.

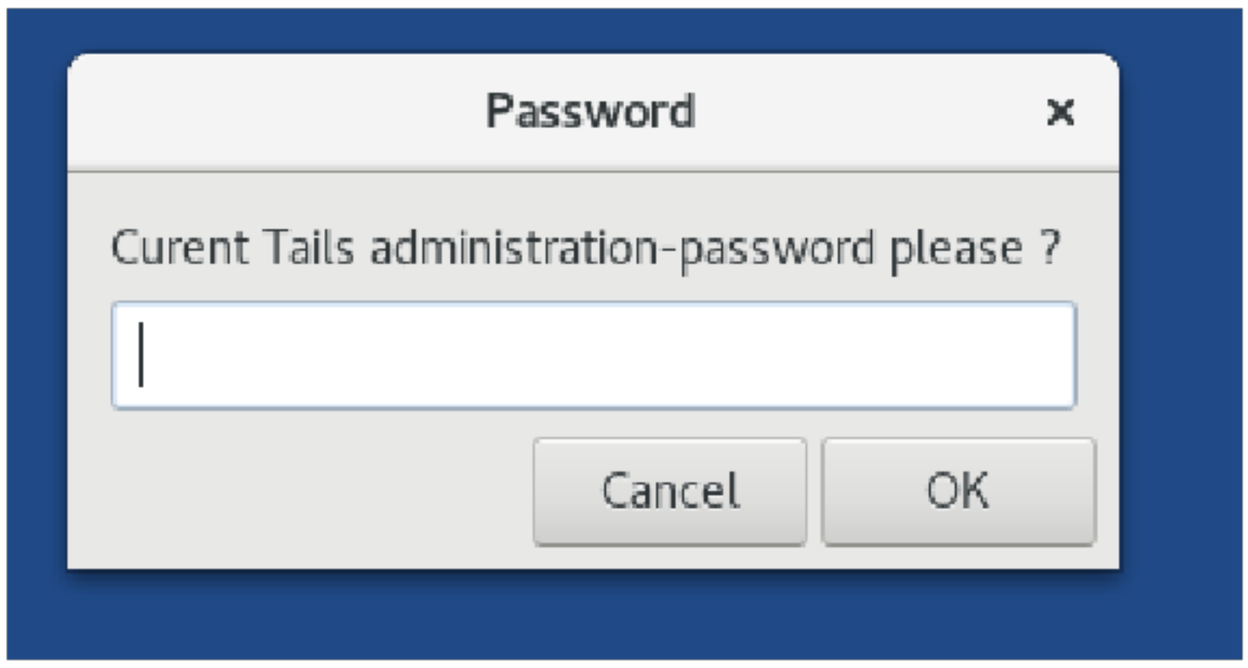
```
cd ~/Persistent/scripts
./swtor-menu.sh
```

Like in the previous chapter, we can start this script over a terminal or over the gui.

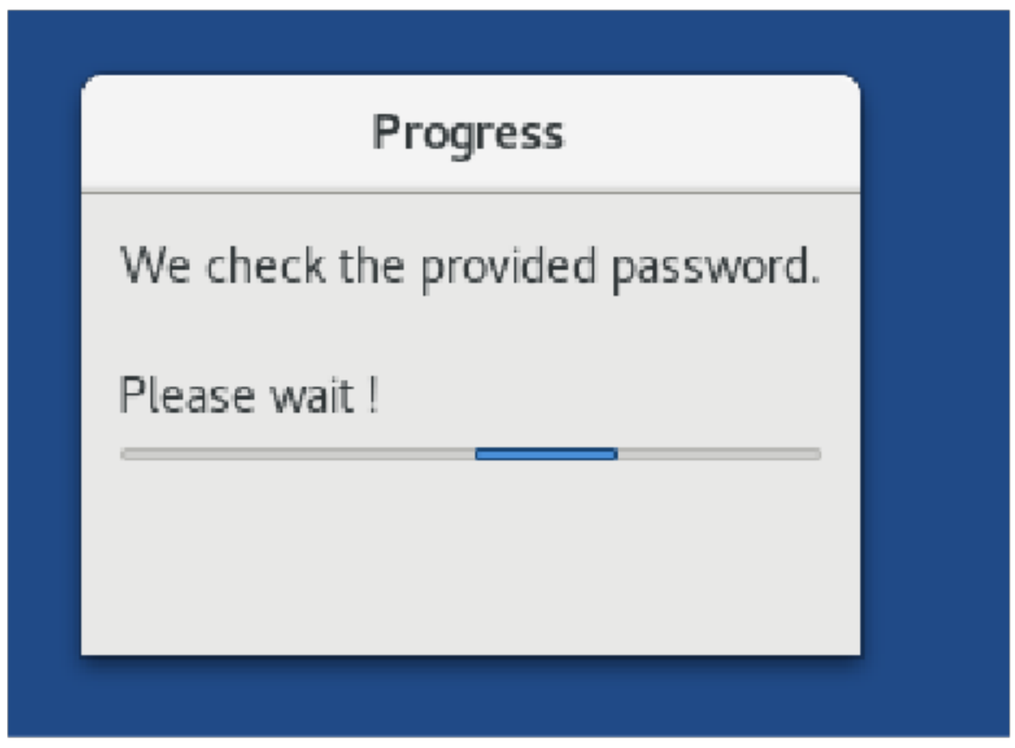
The first window that should appear is this, in the case we found no errors on startop.



The next window will ask for the current administration password.

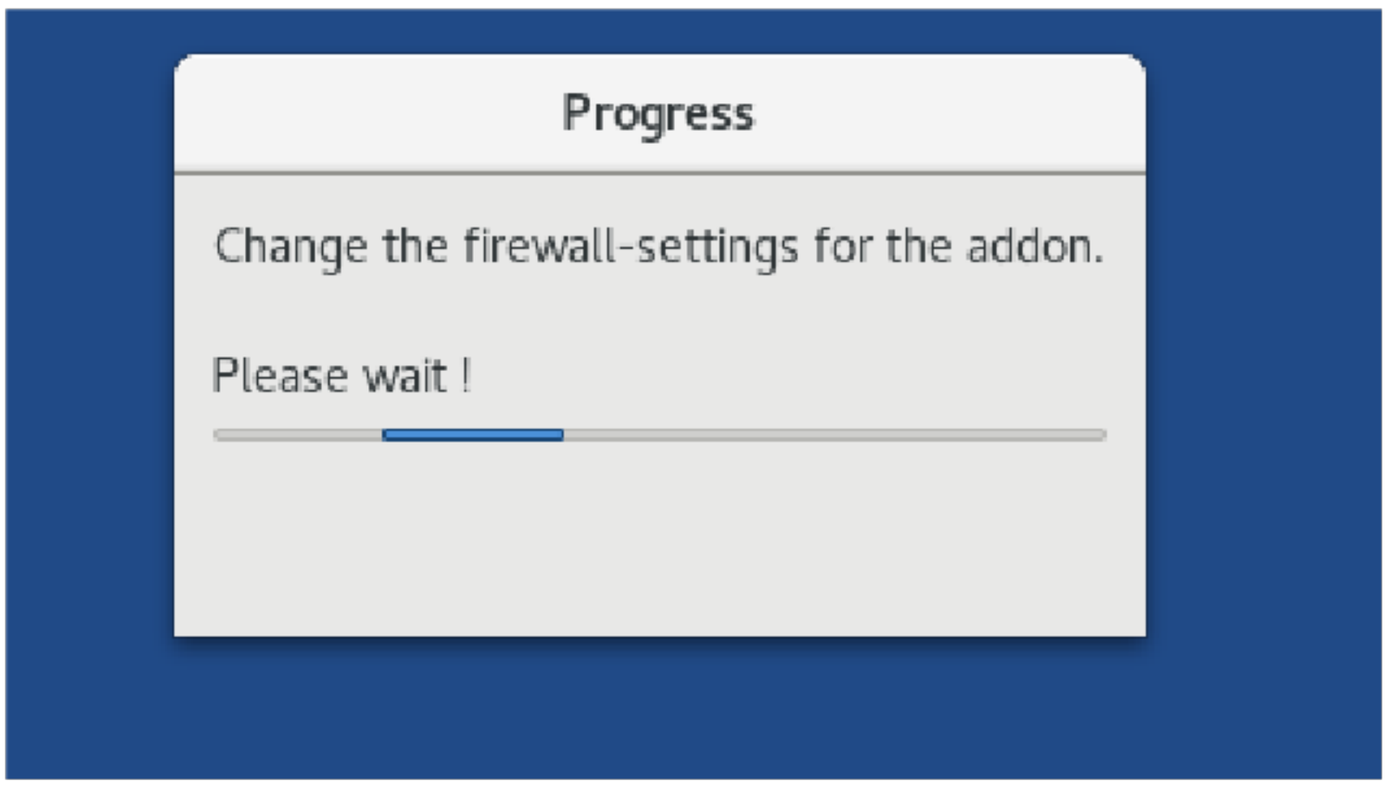


In the next step, the password will be tested.

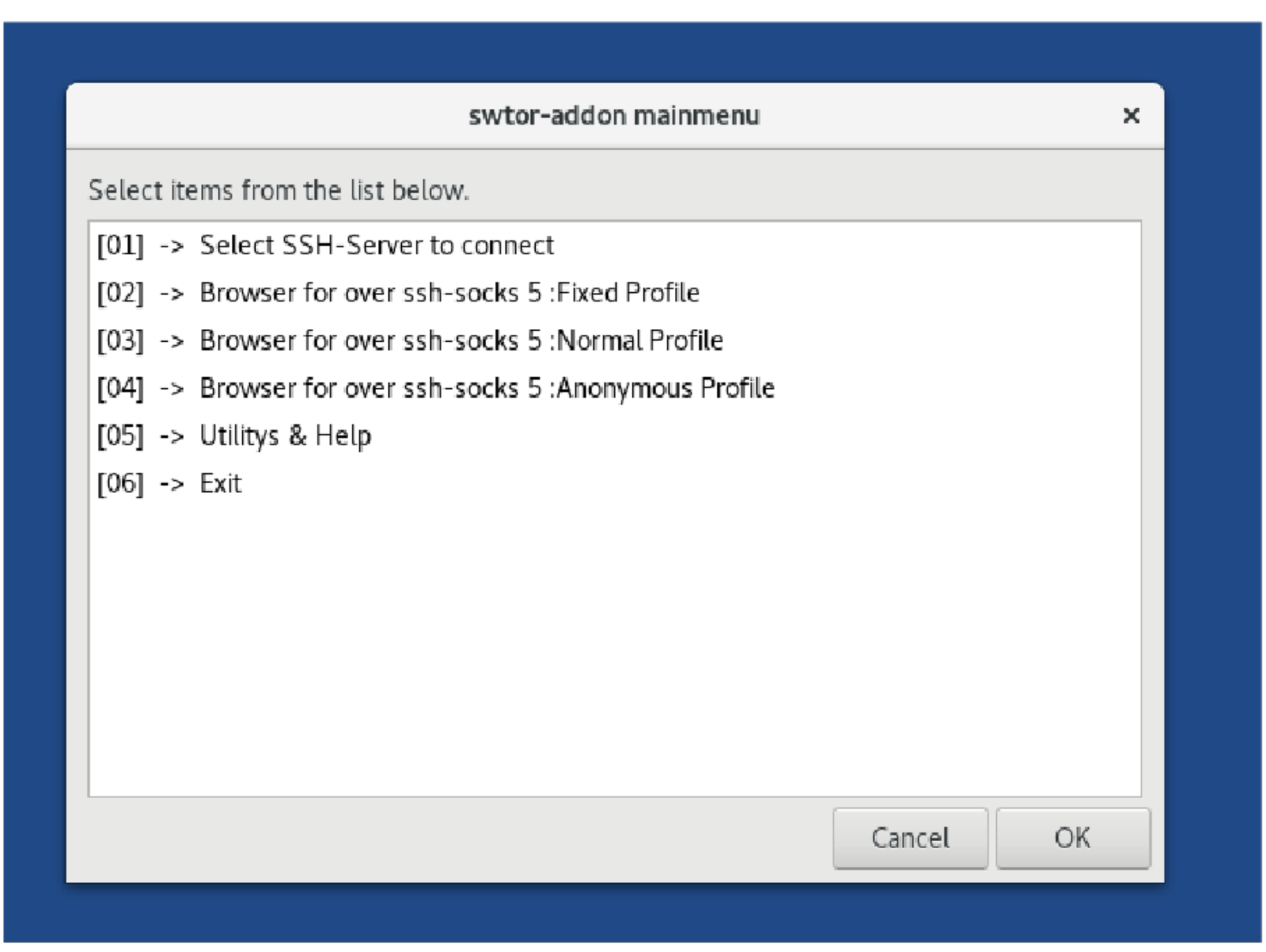


If you provide a wrong or even empty password, the swtor-menu.sh script do quit immediately.

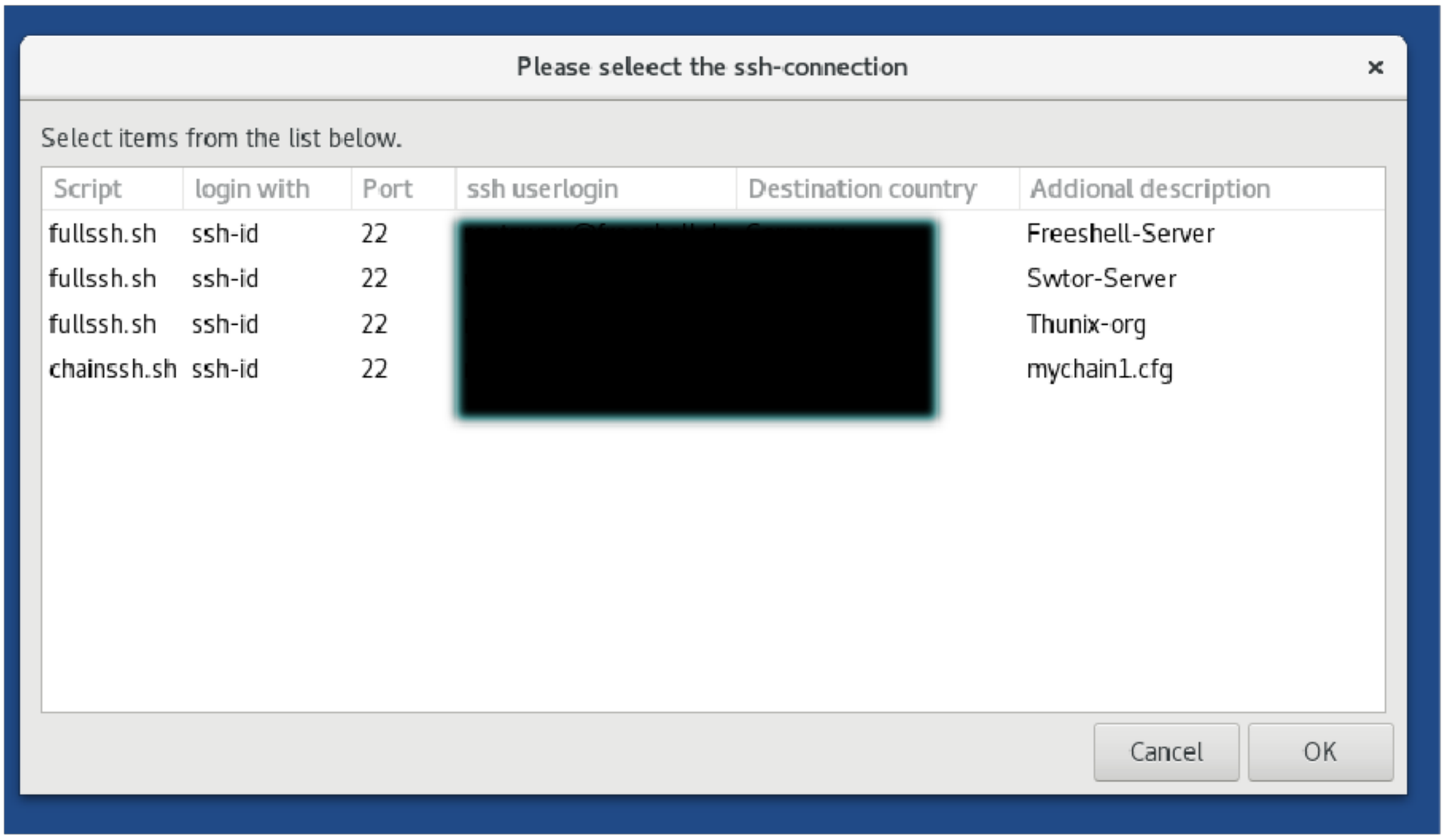
As a final step prior to make the main-menu visible, we do change the firewall settings.



After a few seconds, all preparations are made and the main-menu should appear on the screen. The main-menu looks like here.



The entry 01 of the main-menu is the important one, we should use first. We choose it now !



In the above sample I marked the important area with black color, that shows my own preferred personal ssh-servers. Depending on how many entry's you have inside the configuration file swtorssh.cfg your menu looks may looks almost similar.

You can now select a single connection, that you would like to use. A little windows should appear, that the connection is active.